US 20230020719A1

(19) **United States**

(12) **Patent Application Publication**

Rathod et al.

(10) **Pub. No.: US 2023/0020719 A1**

(43) **Pub. Date: Jan. 19, 2023**

(54) **TECHNOLOGY ENVIRONMENT FOR A SOFTWARE APPLICATION**

(71) Applicant: **Bank of America Corporation**, Charlotte, NC (US)

(72) Inventors: **Chirag Himatlal Rathod**, Mumbai (IN); **Kalpesh Fatehchand Jain**, Thane (IN); **Ramesh Lakshmi Narayanan**, Chennai (IN); **Satish Kumar Kommineni**, Hyderabad (IN); **Venkata Apparao Alla**, Hyderabad (IN); **Sasidhar Purushothaman**, Hyderabad (IN)

(21) Appl. No.: **17/378,024**

(22) Filed: **Jul. 16, 2021**

Publication Classification

(51) **Int. Cl.**

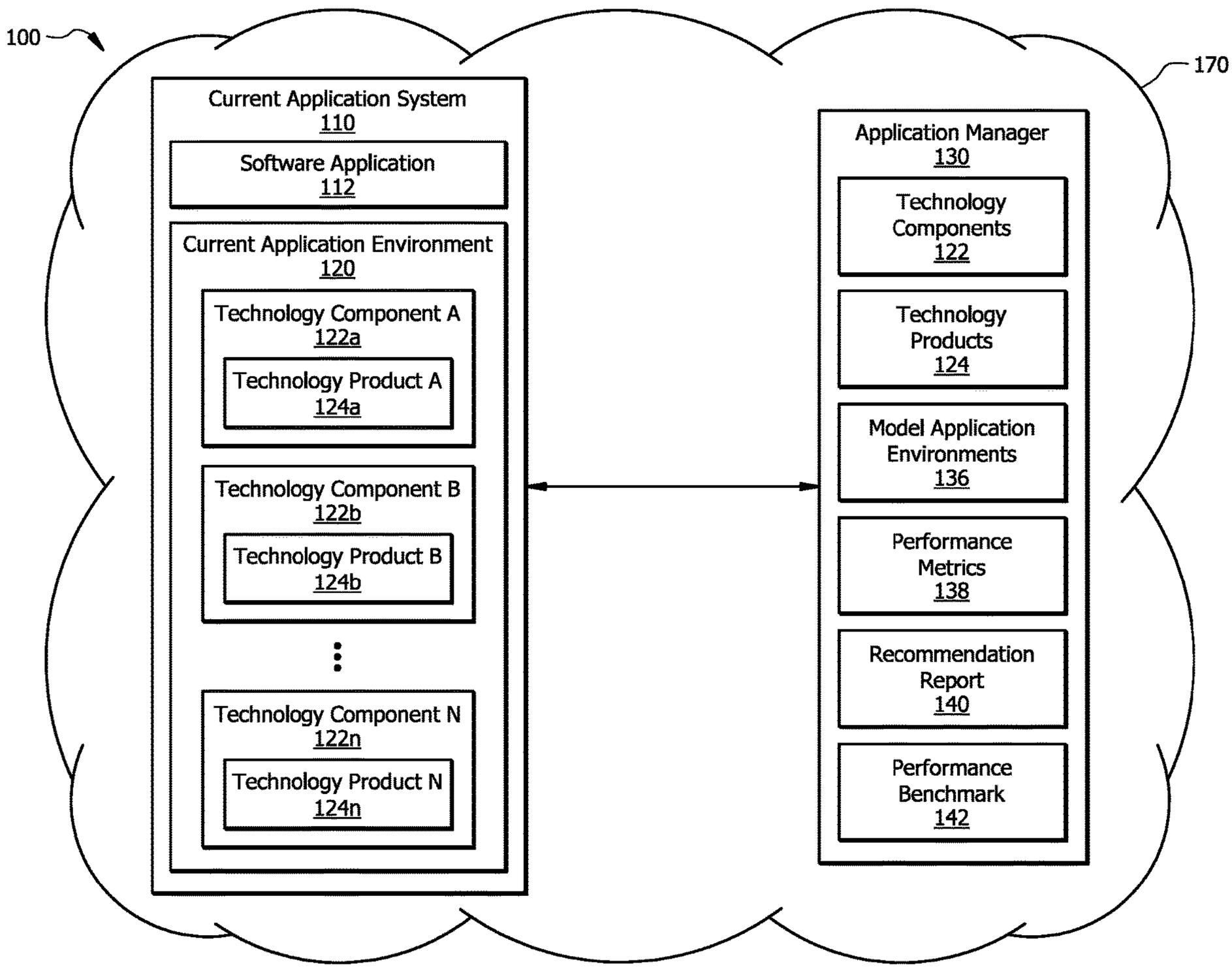
G06F 11/34	(2006.01)
G06F 9/455	(2006.01)

(52) **U.S. Cl.**

CPC ***G06F 11/3495*** (2013.01); ***G06F 9/455*** (2013.01)

(57) **ABSTRACT**

A system is configured to obtain information relating to a current application environment of a software application and build a plurality of model application environments based on the obtained information. The system runs the software application using the current application environment and each of the model application environments. The system collects a plurality of performance metrics related to performance of the software application in the current application environment and each of the model application environments while running in the simulated environment. The system generates a recommendation report based on the performance metrics, wherein the recommendation report comprises a recommendation of a different technology product for at least one of the technology components used in the current application environment, wherein the different technology product is different from a current technology product used for the at least one technology component in the current application environment.



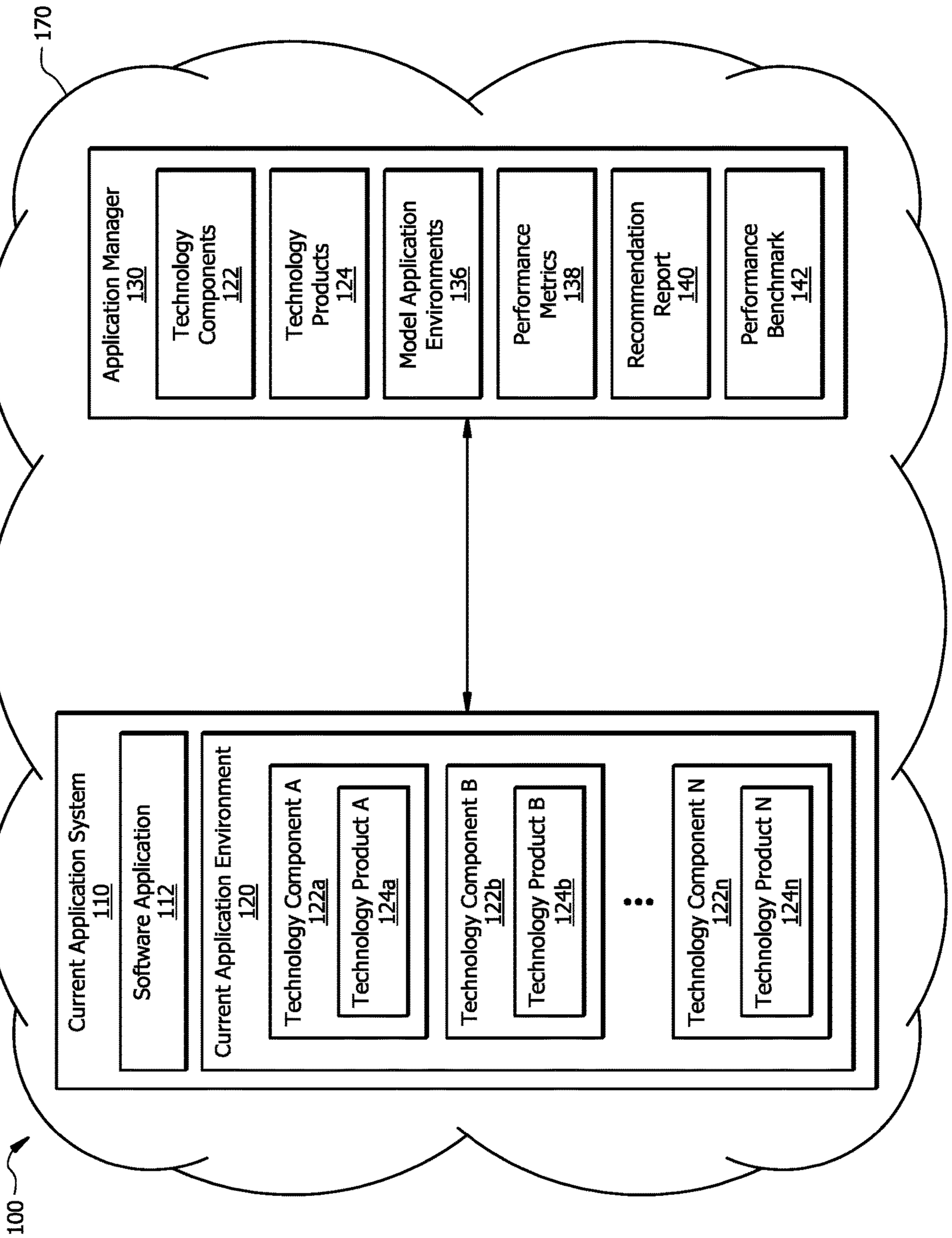

100
170
Application Manager
130
Technology Components
122
Technology Products
124
Model Application Environments
136
Performance Metrics
138
Recommendation Report
140
Performance Benchmark
142
Current Application System
110
Software Application
112
Current Application Environment
120
Technology Component A
122a
Technology Product A
124a
Technology Component B
122b
Technology Product B
124b
Technology Component N
122n
Technology Product N
124n

FIG. 1

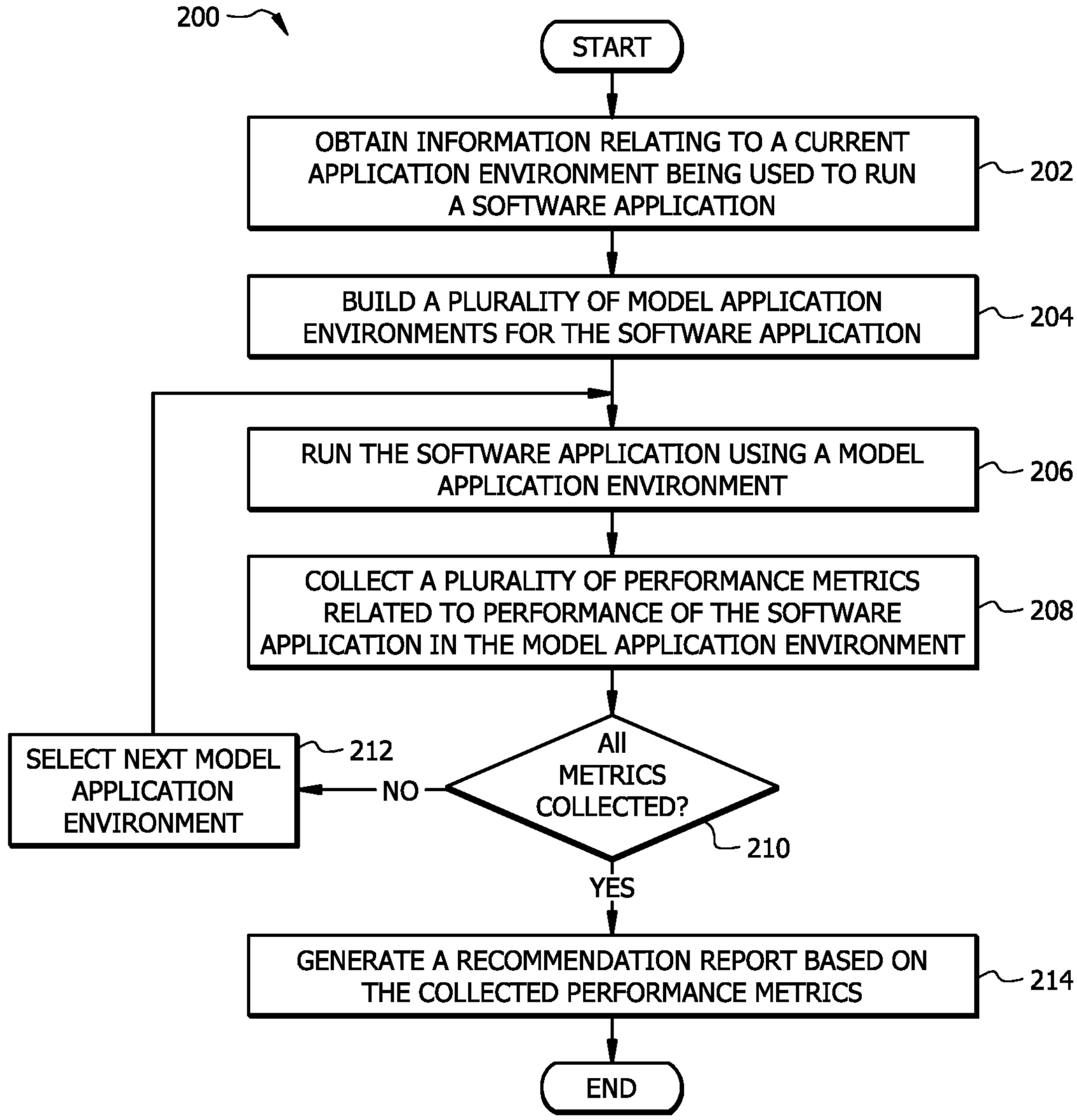

200
START
OBTAIN INFORMATION RELATING TO A CURRENT APPLICATION ENVIRONMENT BEING USED TO RUN A SOFTWARE APPLICATION
202
BUILD A PLURALITY OF MODEL APPLICATION ENVIRONMENTS FOR THE SOFTWARE APPLICATION
204
RUN THE SOFTWARE APPLICATION USING A MODEL APPLICATION ENVIRONMENT
206
COLLECT A PLURALITY OF PERFORMANCE METRICS RELATED TO PERFORMANCE OF THE SOFTWARE APPLICATION IN THE MODEL APPLICATION ENVIRONMENT
208
All METRICS COLLECTED?
210
NO
SELECT NEXT MODEL APPLICATION ENVIRONMENT
212
YES
GENERATE A RECOMMENDATION REPORT BASED ON THE COLLECTED PERFORMANCE METRICS
214
END

FIG. 2

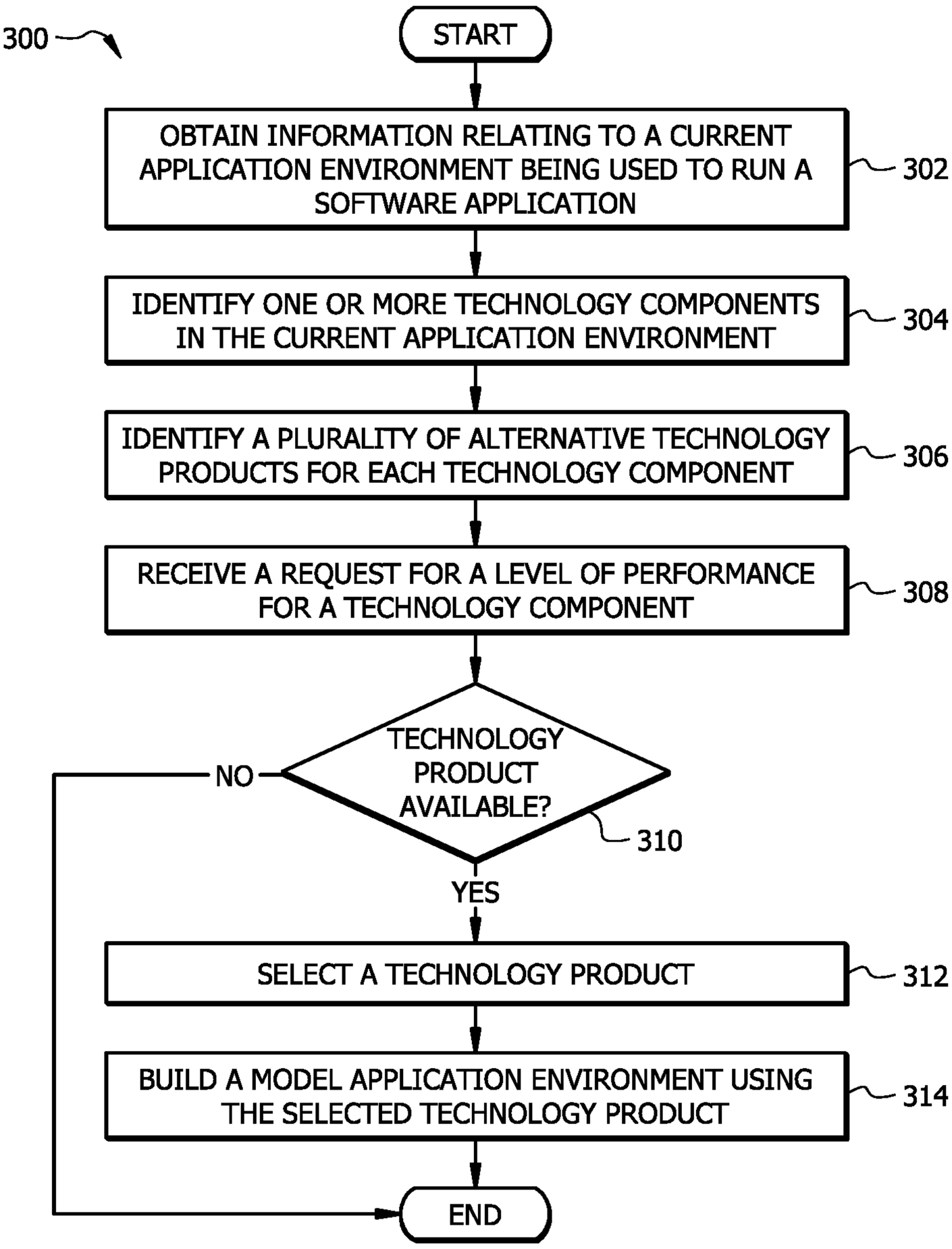

300
START
OBTAIN INFORMATION RELATING TO A CURRENT APPLICATION ENVIRONMENT BEING USED TO RUN A SOFTWARE APPLICATION
302
IDENTIFY ONE OR MORE TECHNOLOGY COMPONENTS IN THE CURRENT APPLICATION ENVIRONMENT
304
IDENTIFY A PLURALITY OF ALTERNATIVE TECHNOLOGY PRODUCTS FOR EACH TECHNOLOGY COMPONENT
306
RECEIVE A REQUEST FOR A LEVEL OF PERFORMANCE FOR A TECHNOLOGY COMPONENT
308
TECHNOLOGY PRODUCT AVAILABLE?
310
NO
YES
SELECT A TECHNOLOGY PRODUCT
312
BUILD A MODEL APPLICATION ENVIRONMENT USING THE SELECTED TECHNOLOGY PRODUCT
314
END

FIG. 3

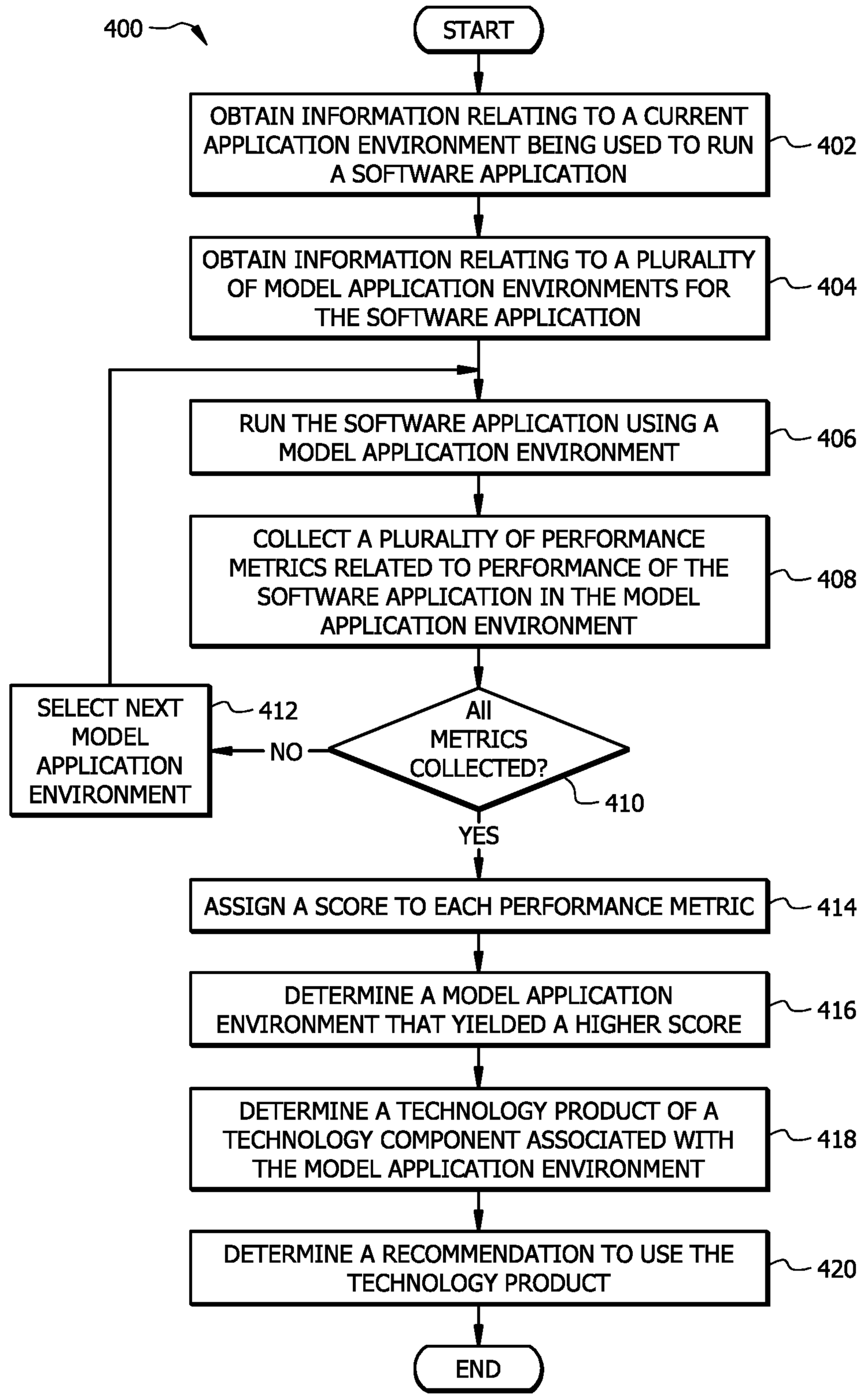

400
START
OBTAIN INFORMATION RELATING TO A CURRENT APPLICATION ENVIRONMENT BEING USED TO RUN A SOFTWARE APPLICATION
402
OBTAIN INFORMATION RELATING TO A PLURALITY OF MODEL APPLICATION ENVIRONMENTS FOR THE SOFTWARE APPLICATION
404
RUN THE SOFTWARE APPLICATION USING A MODEL APPLICATION ENVIRONMENT
406
COLLECT A PLURALITY OF PERFORMANCE METRICS RELATED TO PERFORMANCE OF THE SOFTWARE APPLICATION IN THE MODEL APPLICATION ENVIRONMENT
408
All METRICS COLLECTED?
410
NO
SELECT NEXT MODEL APPLICATION ENVIRONMENT
412
YES
ASSIGN A SCORE TO EACH PERFORMANCE METRIC
414
DETERMINE A MODEL APPLICATION ENVIRONMENT THAT YIELDED A HIGHER SCORE
416
DETERMINE A TECHNOLOGY PRODUCT OF A TECHNOLOGY COMPONENT ASSOCIATED WITH THE MODEL APPLICATION ENVIRONMENT
418
DETERMINE A RECOMMENDATION TO USE THE TECHNOLOGY PRODUCT
420
END

FIG. 4

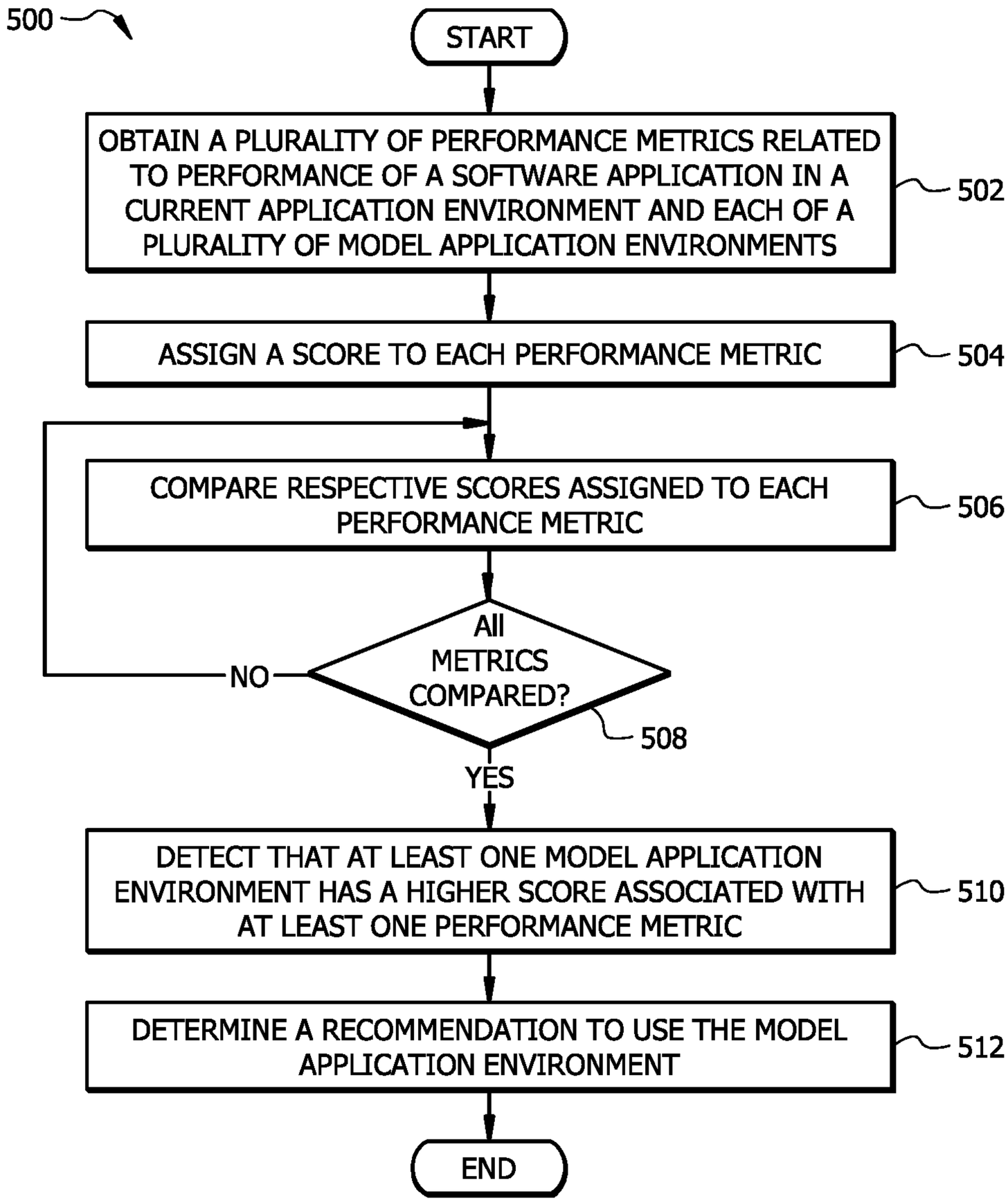

500
START
OBTAIN A PLURALITY OF PERFORMANCE METRICS RELATED TO PERFORMANCE OF A SOFTWARE APPLICATION IN A CURRENT APPLICATION ENVIRONMENT AND EACH OF A PLURALITY OF MODEL APPLICATION ENVIRONMENTS
502
ASSIGN A SCORE TO EACH PERFORMANCE METRIC
504
COMPARE RESPECTIVE SCORES ASSIGNED TO EACH PERFORMANCE METRIC
506
All METRICS COMPARED?
508
NO
YES
DETECT THAT AT LEAST ONE MODEL APPLICATION ENVIRONMENT HAS A HIGHER SCORE ASSOCIATED WITH AT LEAST ONE PERFORMANCE METRIC
510
DETERMINE A RECOMMENDATION TO USE THE MODEL APPLICATION ENVIRONMENT
512
END

FIG. 5

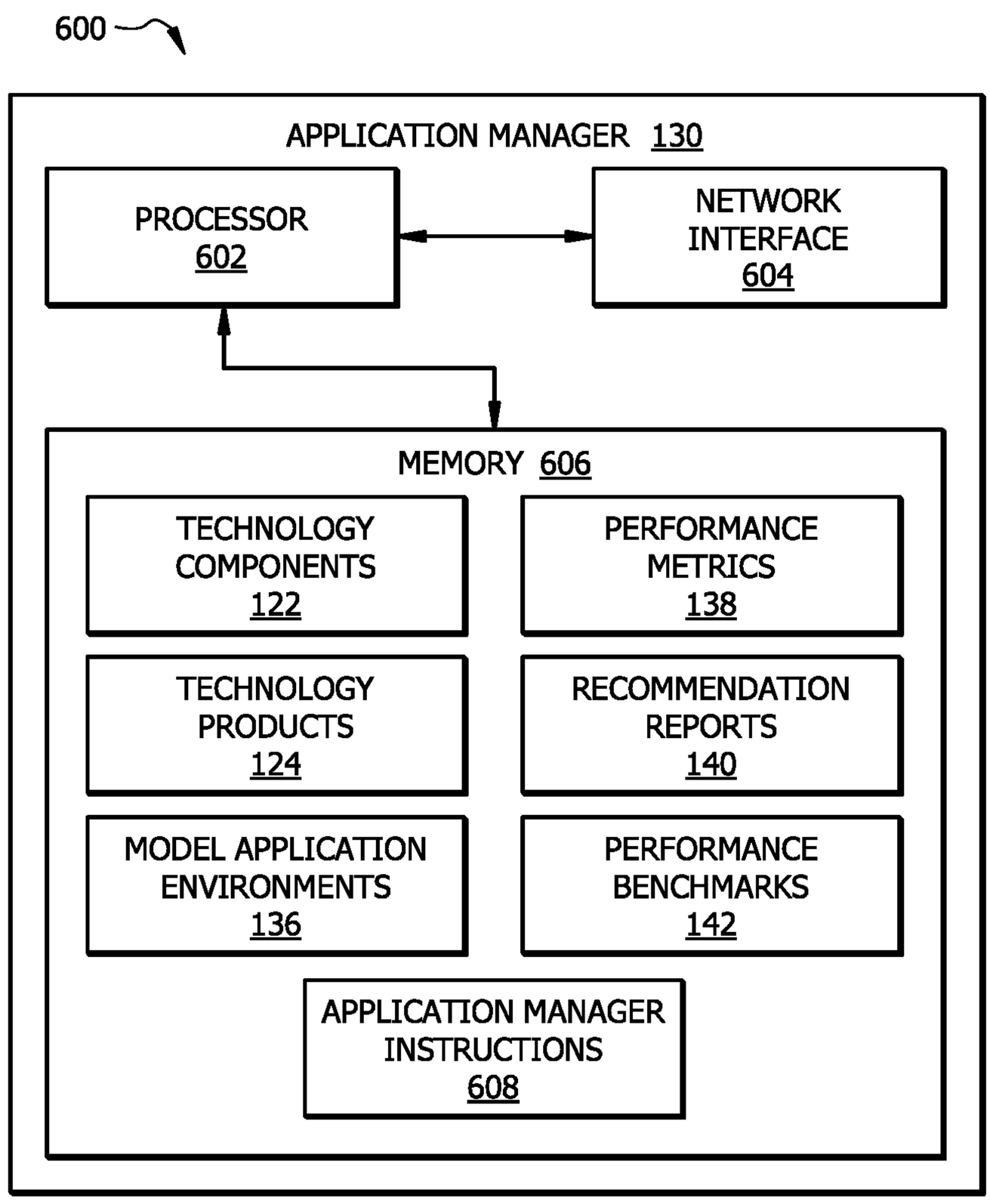

600
APPLICATION MANAGER 130
PROCESSOR 602
NETWORK INTERFACE 604
MEMORY 606
TECHNOLOGY COMPONENTS 122
PERFORMANCE METRICS 138
TECHNOLOGY PRODUCTS 124
RECOMMENDATION REPORTS 140
MODEL APPLICATION ENVIRONMENTS 136
PERFORMANCE BENCHMARKS 142
APPLICATION MANAGER INSTRUCTIONS 608

FIG. 6

TECHNOLOGY ENVIRONMENT FOR A SOFTWARE APPLICATION

TECHNICAL FIELD

[0001] The present disclosure relates generally to managing performance of software implemented systems, and more specifically to improving a technology environment for a software application.

BACKGROUND

[0002] A number of technology products are presently available to implement and run a software application. Currently an application environment for a software application is designed based on one or more desired performance criteria for the application. The application environment is built using a set of technology products that satisfy the performance criteria of the application. The application is expected to behave in a desired manner when deployed in the application environment using the set technology products. However, a technology product may not perform in a desired manner when deployed in the application environment in relation to one or more performance parameters including application performance, deployment strategy, availability issues and other application performance parameters. Thus, there is no way of knowing whether the application will behave in a desired manner when actually deployed in the application environment. For example, an application may be designed for use with a particular type of server expecting 50 to 100 users at any one time. However, when the application is actually deployed, the volume of user traffic may go beyond the expected traffic which the particular server may be incapable of handling. In this case, the application environment may need to be redesigned to suit the changed application requirements. Further, the technology landscape is continuously changing with new and emerging technology products being added regularly and enhancements being added to the current set of technology products.

SUMMARY

[0003] The system and method implemented by the system as disclosed in the present disclosure provide technical solutions to the technical problems discussed above by recommending one or more application environments and/or specific technology products suitable to run a software application. The disclosed system and methods provide several practical applications and technical advantages. For example, the disclosed system provides the practical application of recommending an application environment and/or a technology product that is better suited to the software application as compared to a current application environment of the software application. As disclosed in accordance with embodiments of the present disclosure, an application manager tests different combinations of technology products for the technology components required to run the software application and recommends an alternative combination of technology products for the technology components that is different from the combination of technology products currently used in the current application environment and can yield better performance of the software application corresponding to one or more performance metrics. As disclosed in the following disclosure, the application manager obtains information relating to the technology components of the current application environment and specific technology products used for those technology components in the current application environment. The application manager may additionally obtain information relating to one or more performance requirements of the software application including load performance and processing performance required for the software application. Based on the obtained information, application manager builds a plurality of alternative model application environments that can potentially be used for the software application such that each model application environment includes a different combination of technology products for the technology components needed to run the software application. Application manager runs the software application in each of the model application environments in a simulated environment and collects data relating to a plurality of performance metrics related to performance of the software application in each of the model application environments. Application manager analyzes the performance metrics collected for each model application environment (including the current application environment) and generates a recommendation report based on the analysis. The recommendation report may include a recommendation to use one or more model application environments for the software application that are found to perform better than the current application environment based on the analysis of the performance metrics. Additionally or alternatively, the recommendation report may include recommendations to use individual technology products for one or more technology components that satisfy one or more performance requirements of the software application.

[0004] The disclosed system and methods provide an additional practical application of adjusting the current application environment of the software application from time to time to accommodate changing performance requirements of the software application. As described in accordance with certain embodiments of the present disclosure, the application manager may collect information relating to one or more performance requirements of the software application according to a pre-determined schedule (e.g., periodically). Based on the collected information, application manager may determine whether one or more performance requirements of the software application has changed over time and build a customized application environment for the software application based on the current performance requirements of the software application. This may ensure that the current application environment of the software application evolves to accommodate the changing performance requirements of the software application.

[0005] The disclosed system and methods provide an additional practical application of evolving the current application environment of the software application to accommodate new and emerging technologies. For example, when a new technology product is added for a technology component being using in the current technology environment, the application manager may be configured to assess whether the newly added technology product can be used in an application environment for the software application. For example, when the newly added technology product corresponds to a technology component being used by the current technology environment of the software application, application manager may determine that the newly added technology product may be used in the application environment of the software application. Additionally or alternatively, application manager may determine whether the newly

added technology product improves performance of the software application corresponding to at least one performance metric (e.g., load performance, processing performance etc.). Application manager may be configured to build a new model application environment by selecting the newly added technology product for the respective technology component in response to determining that the newly added technology product can be used in the application environment of the software application and/or that the newly added technology product improves performance of the software application corresponding to at least one performance metric.

[0006] The disclosed system and methods provide an additional practical application of improving performance (e.g., processing performance) of a computing system running the software application by recommending usage of a technology environment and/or technology product that improves the performance of the computing system while running the software application. As disclosed in accordance with certain embodiments of the present disclosure, application manager may maintain information relating to performance benchmarks for each technology product under distinct performance conditions such as user traffic and amount of data being handled by the software application. A performance benchmark relating to a technology product includes a level of performance provided by the technology product corresponding to at least one performance metric. A performance metric, as discussed below, may relate to a load performance and/or processing speed performance of the technology product. For example, a technology product may be associated with high load performance meaning the technology product can handle a large amount of user traffic and/or large amount of data. Another technology product may be associated with high server performance which may mean that the technology product provides fast server response times. Application manager may build custom application environments for the software application by selecting one or more technology products based on their corresponding performance benchmarks, to improve processing performance of the computing system running the software application. For example, application manager may include a technology product in the application environment that improves the response time of the computing system. running the software. Similarly, application manager may include a technology product in the application environment that can handle a higher load (e.g., user traffic, data load etc.) to avoid the application server from freezing and/or crashing, thus improving the performance of the application server.

BRIEF DESCRIPTION OF THE DRAWINGS

[0007] For a more complete understanding of this disclosure, reference is now made to the following brief description, taken in connection with the accompanying drawings and detailed description, wherein like reference numerals represent like parts.

[0008] FIG. **1** is a schematic diagram of an example data processing system, in accordance with certain embodiments of the present disclosure;

[0009] FIG. **2** is a flowchart of an example method for processing an application environment for a software application, in accordance with certain embodiments of the present disclosure;

[0010] FIG. **3** is a flowchart of an example method for building application environments for a software application, in accordance with certain embodiments of the present disclosure;

[0011] FIG. **4** is a flowchart of an example method for improving an application environment for a software application, in accordance with certain embodiments of the present disclosure;

[0012] FIG. **5** is a flowchart of an example method for analyzing performance metrics collected for model application environments of a software application, in accordance with certain embodiments of the present disclosure; and

[0013] FIG. **6** illustrates an example schematic diagram of the application manager illustrated in FIG. **1**, in accordance with one or more embodiments of the present disclosure.

DETAILED DESCRIPTION

System Overview

[0014] FIG. **1** is a schematic diagram of an example data processing system **100**, in accordance with certain embodiments of the present disclosure.

[0015] As shown in FIG. **1**, data processing system **100** includes a current application system **110** and an application manager **130**, each connected to a network **170**. The network **170**, in general, may be a wide area network (WAN), a personal area network (PAN), a cellular network, or any other technology that allows devices to communicate electronically with other devices. In one or more embodiments, the network **170** may be the Internet.

[0016] The current application system **110** may run a software application **112** in a current application environment **120**. The current application environment **120** may use technology components (shown as **122***a*-**122***n*), wherein each technology component **122***a*-**122***n* uses a respective technology product (shown as **124***a*-**124***n*). In one example use case, current application system **110** may be representative of a production server of an organization that runs the software application **112** in the current application environment **120**. The production server may be configured to be accessed by several users desiring to use the functionality implemented by the software application **112**. Each technology component **122***a*-**122***n* may refer to a type of hardware or software tool used to implement, run and/or support the software application **112**. For example, a technology component may include, but is not limited to, a middleware component, an operating system component, a database engine component, an implementation technology component (e.g., coding language used to implement the software application **112**), a cloud provider component, a network technology component (e.g., WAN, LAN, VPN etc.), a virtualization technique component and a server less computing technology component. A technology product **124***a*-**124***n* may refer to a specific hardware or software tool/technology used for a corresponding technology component. For example, Apache Tomcat® can be a technology product used as a middleware component. Thus, the term "technology component" may generally refer to a type of technology product, and the term "technology product" may refer to a specific hardware or software tool used for a respective technology component. In one example case, current application environment **120** may include three technology components **122***a*, **122***b* and **122***c* using respective technology products **124***a*, **124***b* and **124***c*. Technology component **122***a*

may be a middleware component that uses Apache Tomcat server as the specific technology product **124***a*. Technology component **122***b* may be a database engine component that uses SQL server as the specific technology product **124***b*. Technology component **122***c* may be a virtualization technique component that uses VMware virtual machine as the specific technology product **124***c*.

[0017] Several alternative technology products **124** are typically available corresponding to each technology component **122***a*-**122***n*. For example, technology product alternatives for the operating system (OS) component may include, but are not limited to, Windows OS, Linux OS, Redhat OS and Fedora OS. Technology product alternatives for the implementation technology component may include, but are not limited to, C++, Java, C#, Python and JavaScript. Technology product alternatives for the middleware technology component may include, but are not limited to, Apache Tomcat server, JBoss server and Oracle server. Technology product alternatives for the cloud component may include, but are not limited to, Microsoft Azure, Amazon webservices and Google cloud platform. Technology product alternatives for the network component may include, but are not limited to, Local Area Network (LAN), Wireless (WLAN), Wide Area Network (WAN) and Virtual Private Network (VPN). Technology product alternatives for the virtualization technique component may include, but are not limited to, VMware, Microsoft Hyper-V, Openstack and Xen.

[0018] As described above, the technology products **124***a*-**124***n* used for the respective technology components **122***a*-**122***n* in the current application environment **120** may not be the most optimal for the software application **112**. That is, the current application environment **120** may not always yield the performance desired for the software application **112**.

[0019] Application manager **130** is configured to test different combinations of technology products **124** for the technology components **122***a*-**122***n* required to run the software application **112** and recommend an alternative combination of technology products **124** for the technology components **122***a*-**122***n* that is different from the combination of technology products **124***a*-**124***n* currently used in the current application environment **120** and can yield better performance of the software application **112** corresponding to one or more performance metrics.

[0020] Application manager **130** may be configured to store information relating to a plurality of alternative technology products **124** that can be used for each of a plurality of technology components **122**. Application manager **130** may be configured to build a plurality of model application environments **136** such that each of the model application environments **136** includes a different combination of the technology products **124** used for the technology components **122**. Application manager **130** may be configured to obtain information relating to the current application environment **120** being used for the software application **112**. In one embodiment, application manager **130** may use an application connector to obtain the information relating to the current application environment **120**. The application connector may be a software agent designed to run on the current application system **110** and extract information relating to the current application environment **120** from the current application system **110**. The information extracted by the application connector may include information relating to the particular technology components **122***a*-**122***n* of the current application environment **120** and the specific technology products **124***a*-**124***n* being used for each technology component **122***a*-**122***n*. In one embodiment, the information extracted by the application connecter may include information relating to one or more performance requirements of the software application with regard to one or more performance metrics including load performance and processing performance. As described below, the information relating to the performance requirements of the software application may be used by application manager **130** to recommend one or more model application environments **136** and/or technology products that suit the performance requirements.

[0021] Based on the information obtained from the current application system **110**, application manager **130** may be configured to build a plurality of alternative model application environments **136** that can potentially be used for the software application **112**. For example, based on the information obtained from the current application system **110**, application manager **130** may determine the technology components **122***a*-**122***n* that need to be incorporated into an application environment meant for the software application **112**. Once, the technology components **122***a*-**122***n* are identified, application manager **130** may identify the plurality of alternative technology products **124** available for each of the technology components **122***a*-**122***n*. The alternative technology products **124** identified for each technology component **122***a*-**122***n* may include the current technology product **124***a*-**124***n* being used in the current application environment **120** for the technology components **122***a*-**122***n*. Application manager **130** may be configured to build a plurality of model application environments **136** for the software application **112** by selecting for each model application environment **136** a different combination of the technology products **124** for the technology components **122***a*-**122***n*. That is, each model application environment **136** is built to use a combination of technology products **124** that is different from the combinations of technology products **124** used for other model application environments **136** built for the software application **112**. In one embodiment, application manager **130** also builds a copy of the current application environment **120** for comparing with other model application environments **136**.

[0022] For example, when the technology components **122***a*-**122***n* of the current application environment **120** includes a middleware component, a database engine component and a virtualization technique component, application manager **130** may identify alternative technology products for each of a middleware component, a database engine component and a virtualization technique component. Application manager **130** may build a plurality of potential model application environments **136** for the software application **112** by using different combinations of technology products **124** for the middleware component, database engine component and virtualization technique component. For example, application manager **130** may build each model application environment by selecting a different alternative technology product **124** for at least one of the technology components **122***a*-**122***n*.

[0023] Application manager **130** may be configured to automatically configure each of the model application environments **136** to suit the software application **112**. The application manager **130** may be configured to obtain infor-

mation relating to how the software application **112** is configured with respect to one or more parameters (e.g., logging-in, load balancing, database details etc.) in the current application environment **120**. Application manager **130** may configure each of the model application environments **136** for the software application **112**, based on the configurations in the current application environment **120**, for example, to match the configurations in the current application environment **120**. Application manager **130** may also configure one or more parameters of a model application environment by modifying the configuration of the parameters in the current application environment **120** to suit the model application environment **136**.

[0024] Application manager **130** may maintain information relating to performance benchmarks **142** for each technology product **124** under distinct performance conditions such as user traffic and amount of data being handled. A performance benchmark **142** relating to a technology product **124** includes a level of performance provided by the technology product **124** corresponding to at least one performance metric **138**. A performance metric **138**, as discussed below, may relate to a load performance and/or processing speed performance of the technology product **124**. For example, a technology product **124** may be associated with high load performance meaning the technology product **124** can handle a large amount of user traffic and/or large amount of data. Another technology product **124** may be associated with high server performance which may mean that the technology product **124** provides fast server response times. The performance benchmarks **142** of the technology products **124** may allow the application manager **130** to build custom model application environments by selecting one or more technology products **124** that satisfy specific performance requirements of the software application **112**. For example, application manager **130** may be configured to obtain information relating to at least one performance requirement of the software application **112** corresponding to a technology component **122**. Application manager **130** may be configured to determine a technology product **124** available for the technology component **122** and that satisfies the performance requirement, based on the performance benchmark **142** stored for the technology product **124**. Application manager **130** may be configured to build a model application environment **136** for the software application **112** by selecting the technology product **124** for the technology component **122**.

[0025] Application manager **130** may be configured to evolve to accommodate new and emerging technologies. For example, when a new technology product **124** is added for a technology component **122**, application manager **130** may be configured to assess whether the newly added technology product **124** can be used in an application environment for the software application **112**. For example, when the newly added technology product **124** corresponds to a technology component **122***a*-**122***n* being used by the current technology environment **120** of the software application **112**, application manager **130** may determine that the newly added technology product **124** may be used in the application environment of the software application **112**. Additionally or alternatively, application manager **130** may be configured to determine whether the newly added technology product **124** satisfies one or more performance requirements of the software application **112**. Application manager **130** may be configured to build a new model application environment **136** by selecting the newly added technology product **124** for the respective technology component **122**. Application manager **130** may build the new model application environment **136** in response to determining that the newly added technology product **124** can be used in the application environment of the software application **112** and/or that the newly added technology product **124** satisfies at least one requirements of the software application **112**. In one embodiment, application manager **130** may be configured to collect information relating to one or more performance requirements of the software application **112** according to a pre-determined schedule (e.g., periodically). Application manager **130** may be configured to detect whether the performance requirements of the software application **112** has changed and build a customized application environment for the software application **112** based on current performance requirements of the software application **112**. This may ensure that the current application environment **120** of the software application **112** evolves to accommodate the changing performance requirements of the software application **112**.

[0026] Application manager **130** may be configured to run the software application **112** in each of the model application environments **136** in a simulated environment. One of the model application environments **136** may be a copy of the current application environment **120**. Application manager **130** may include a simulator logic that can be configured to host and deploy each of the model application environments **136** (including the current application environment **120**) and run the software application **112** in each of the model application environments **136**. Application manager **130** may be configured to collect data relating to a plurality of performance metrics **138** related to performance of the software application in each of the model application environments **136** while running in the simulated environment. The plurality of performance metrics **138** may include, but are not limited to, one or more of application server performance, performance related to type of data handled by the software application, database performance, data security analysis, input and output analysis and deployment on cloud platforms. Application server performance may relate to response time of the application server at different levels of load (e.g., user traffic, amount of data etc.). Performance data related to the type of data handled by the software application **112** may include information relating to whether the software application **112** handles sensitive data needing a higher level of security and whether a model application environment **136** can handle sensitive data. This performance metric may allow the application manager **130** to recommend secure technology products **124** that can securely handle sensitive user data, such as private cloud product or a virtual machine where the virtualization is done at the hardware level to provide maximum security. Database performance may relate to database response times. Data relating to data security analysis may include security vulnerabilities (e.g., relating to data security) of the software application **112** in particular model application environments **136**. Application manager **130** scans the software application **112** for known vulnerabilities in each model application environment **136** and for potential threats to the application **112**. Application manager **130** may determine based on this performance metric whether a particular model application environment **136** is vulnerable to attacks. Data related to deployment on cloud platforms may include

information relating to whether the software application **112** can be deployed on cloud platforms. For example, application manager **130** may be configured to analyze whether the software application **112** is suitable for deploying on a cloud platform based on running the software application **112** in model application environments **136** including cloud deployments. This performance metric may help the application manager **130** determine whether the software application **112** can be deployed on cloud platforms or is more suitable for a private in-house server or virtual machine deployment.

[0027] To measure the performance of the software application **112** in each model application environment **136**, application manager **130** may be configured to run one or more tests on each model application environment **136** while the simulator logic is running the software application **112** in the model application environment **136**. Application manager **130** may collect one or more of the performance metrics **138** described above corresponding to each model application environment **136** as a result of conducting the one or more tests. The tests may include load testing to test the software application **112** at different amounts of load (e.g., user traffic, amount of data etc.). For example, load testing may include simulating simultaneous logins of different numbers of users using different amounts of test data. Additionally or alternatively, the tests may include performance testing to test the overall performance of a model application environment **136** and collect data on performance metrics **138** including availability, response time and stability of the software application **112**. In one example, for each model application environment **136**, application manager **130** carries out the testing in multiple cycles with different amounts of load. For example, each model application environment **136** is tested initially with 10 k simultaneous user logins and 100 GB of test data. This test cycle is then repeated three times with a replication load factor of 2×. This means with each test cycle, the load applied is two times that applied in the previous test cycle. In one embodiment, the load size may be customized based on the requirements of the software application **112**. In an additional embodiment, application manager **130** tests each model application environment **136** for security vulnerabilities (e.g., data security) and collects data relating to how vulnerable the model application environment **136** is to potential security threats.

[0028] Application manager **130** may be configured to analyze the performance metrics **138** collected for each model application environment **136** (including the current application environment **120**) and generate a recommendation report **140** based on the analysis. The recommendation report **140** may include a recommendation to use one or more model application environments **136** for the software application **112** that are found to perform better than the current application environment **120** based on the analysis of the performance metrics **138**. Additionally or alternatively, the recommendation report **140** may include recommendations to use individual technology products **124** for one or more technology components **122** that satisfy one or more performance requirements of the software application **112**.

[0029] In order to analyze the performance metrics **138**, application manager **130** may be configured to segregate the performance metrics **138** collected for each model application environment **136** including the current application environment **120**. The segregating may include grouping together data collected for each performance metric **138** for all the model application environments **136**. For example, data relating to application server performance metric collected for all model application environments **136** may be grouped together. Similarly, data relating to database performance metric collected for all model application environments **136** may be grouped together. Application manager **130** may compare the model application environments **136** based on each performance metric **138** collected for the respective model application environments **136**. For example, the application server performance metric of all model application environments **136** may be compared separately. Similarly, the database performance metric of all model application environments **136** may be compared separately. Application manager **130** may recommend one or more model application environments **136** and/or individual technology products **124** based on results of the comparison. In one or more embodiments, for each model application environment **136** (including the current application environment **120**), application manager **130** assigns a performance score to each performance metric **138** collected for the model application environment **136**. Application manager **130** may be configured to assign a higher score for a higher level of performance related to a performance metric **138**. For example, if a first model application environment **136** is found to be able to handle **500** simultaneous user logins at one time and a second model application environment **136** is found to be able to handle **10***k* simultaneous user logins at one time, the application performance metric for the second model application environment **136** is assigned a higher performance score than the application performance metric for the first model application environment **136**. Application manager **130** may be configured to compare the respective scores of each performance metric **138** collected for several model application environments **136**. Application manager **130** may be configured to recommend a model application environment **136** that is found to have a higher score with regard to one or more performance metrics **138** as compared to the respective scores of the one or more performance metrics **138** collected for other model application environments **136**. Following the above example, application manager **130** may recommend the second model application environment **136** for the software application **112** over the first model application environment **136** based on the higher score associated with the application server performance metric collected for the second model application environment **136**.

[0030] In one or more embodiments, application manager **130** may be configured to compare the total performance scores of multiple performance metrics **138** collected for each model application environment **136** and recommend a model application environment **136** that has the highest total performance score for the performance metrics **138**. For example, application manager **130** may compare the total performance scores of all performance metrics **138** collected for each model application environment **136** and recommend a model application environment **136** that has the highest total performance score of for the performance metrics **138**.

[0031] In one or more embodiments, application manager **130** may be configured to sort the performance metrics **138** collected for each model application environment **136** (including the current application environment **120**) based on one or more requirements of the software application **112**,

for example, by placing a higher priority on performance metrics **138** that are associated with the one or more requirements of the software application **112**. Application manager **130** may be configured to recommend one or more model application environments **136** and or individual technology products **124** that have higher scores recorded for the prioritized performance metrics **138** associated with the one or more requirements of the software application **112**, regardless of scores recorded for other performance metrics **138**. For example, the software application **112** may require high database performance (e.g., quick response times) and high data security as a result of handling sensitive user data. Accordingly, application manager **130** may prioritize the database performance metric and the security performance metric over other performance metrics **138** collected for the model application environments **136**. In this case, application manager **130** recommends one or more model application environments **136** that were found to have the highest scores for the database performance metric and the security performance metric. Application manager **130** may be configured to obtain the one or more requirements of the software application **112** in response to receiving a request for the one or more requirements or by automatically determining the one or more requirements by analyzing one or more performance metrics **138** of the software application **112**. For example, based on analyzing the software application **112** while running using one or more model application environments **136**, application manager may determine that the software application is vulnerable to cyber attacks. This detection may prompt the application manager **130** to prioritize the data security metric for recommending model application environments **136** that were found to have the highest scores related to the data security metric.

[0032] In one embodiment, application manager **130** may compare one or more model application environments **136** with the current application environment **120** based on the performance scores of a prioritized performance metric **138** (e.g., database performance, application server performance etc.). If application manager **130** determines that a model application environment **136** has a higher score for the performance metric **138** as compared to the respective score of the performance metric **138** collected for the current application environment **120**, application manager **130** may be configured to determine at least one technology product **124** used for a corresponding technology component **122** associated with the performance metric **138** in the determined model application environment **136** that is different from the current technology product **124** used for the technology component **122** in the current application environment **120**. Application manager **130** may be configured to recommend using the determined at least one technology product **124** for the technology component **122** to achieve a higher level of performance related to the prioritized performance metric **138**.

[0033] In an additional or alternative embodiment, application manager **130** may be configured to receive a request for a higher level of performance related to a performance metric **138** associated with at least one technology component **122** as compared to a current level of performance related to the performance metric **138** in the current application environment **120**. In response, application manager **130** may be configured to compare one or more model application environments **136** with the current application environment **120** based on the performance scores of the requested performance metric **138**. Based on the results of the comparison, application manager **130** may be configured to determine a model application environment **136** that yielded a higher score for the requested performance metric **138** compared to the score of the performance metric **138** in the current application environment **120**. Application manager **130** may determine a technology product **124** used for the at least one technology component **122** in the determined model application environment **136** and recommends using the determined technology product **124** for the at least one technology component **122** to achieve the requested higher level of performance related to the performance metric **138**.

[0034] FIG. **2** is a flowchart of an example method **200** for processing an application environment for a software application, in accordance with certain embodiments of the present disclosure. Method **200** may be performed by application manager **130** as shown in FIG. **1** and described above.

[0035] At step **202**, application manager **130** obtains information relating to a current application environment **120** being used to run a software application **112**. As described above, current application system **110** may run the software application **112** in the current application environment **120**. The current application environment **120** may use technology components (shown as **122***a*-**122***n*), wherein each technology component **122***a*-**122***n* uses a respective technology product (shown as **124***a*-**124***n*). The technology components **122***a*-**122***n* may include one or more hardware technology components, one or more software technology components or a combination there of. For example, each technology component **122***a*-**122***n* may refer to a type of hardware or software tool used to implement, run and/or support the software application **112**. For example, a technology component may include, but is not limited to, a middleware component, an operating system component, a database engine component, an implementation technology component (e.g., coding language used to implement the software application **112**), a cloud provider component, a network technology component (e.g., WAN, LAN, VPN etc.), a virtualization technique component and a server less computing technology component. A technology product **124***a*-**124***n* may refer to a specific hardware or software tool/technology used for a corresponding technology component. For example, Apache Tomcat® can be a technology product used as a middleware component. Thus, the term "technology component" may generally refer to a type of technology product, and the term "technology product" may refer to a specific hardware or software tool used for a respective technology component. In one example case, current application environment **120** may include three technology components **122***a*, **122***b* and **122***c* using respective technology products **124***a*, **124***b* and **124***c*. Technology component **122***a* may be a middleware component that uses Apache Tomcat server as the specific technology product **124***a*. Technology component **122***b* may be a database engine component that uses SQL server as the specific technology product **124***b*. Technology component **122***c* may be a virtualization technique component that uses VMware virtual machine as the specific technology product **124***c*.

[0036] Application manager **130** may obtain information relating to the current application environment **120** being used for the software application **112**. In one embodiment, application manager **130** may use an application connector to obtain the information relating to the current application environment **120**. The application connector may be a

software agent designed to run on the current application system **110** and extract information relating to the current application environment **120** from the current application system **110**. The information extracted by the application connector may include information relating to the particular technology components **122***a*-**122***n* of the current application environment **120** and the specific technology products **124***a*-**124***n* being used for each technology component **122***a*-**122***n*. In one embodiment, the information extracted by the application connecter may include information relating to one or more performance requirements of the software application with regard to one or more performance metrics **138** including load performance and processing performance. As described below, the information relating to the performance requirements of the software application may be used by application manager **130** to recommend one or more model application environments **136** and/or technology products **124** that suit the performance requirements.

[0037] At step **204**, application manager **130** builds a plurality of model application environments **136** for the software application **112**, based on the information obtained relating to the current application environment **120**.

[0038] As described above, several alternative technology products **124** are typically available corresponding to each technology component **122***a*-**122***n*. For example, technology product alternatives for the operating system (OS) component may include, but are not limited to, Windows OS, Linux OS, Redhat OS and Fedora OS. Technology product alternatives for the implementation technology component may include, but are not limited to, C++, Java, C#, Python and JavaScript. Technology product alternatives for the middleware technology component may include, but are not limited to, Apache Tomcat server, JBoss server and Oracle server. Technology product alternatives for the cloud component may include, but are not limited to, Microsoft Azure, Amazon webservices and Google cloud platform. Technology product alternatives for the network component may include, but are not limited to, Local Area Network (LAN), Wireless (WLAN), Wide Area Network (WAN) and Virtual Private Network (VPN). Technology product alternatives for the virtualization technique component may include, but are not limited to, VMware, Microsoft Hyper-V, Openstack and Xen.

[0039] As described above, the technology products **124***a*-**124***n* used for the respective technology components **122***a*-**122***n* in the current application environment **120** may not be the most optimal for the software application **112**. That is, the current application environment **120** may not always yield the performance desired for the software application **112**.

[0040] Based on the information obtained from the current application system **110**, application manager **130** may build a plurality of alternative model application environments **136** that can potentially be used for the software application **112**. For example, based on the information obtained from the current application system **110**, application manager **130** may determine the technology components **122***a*-**122***n* that need to be incorporated into an application environment meant for the software application **112**. Once, the technology components **122***a*-**122***n* are identified, application manager **130** may identify the plurality of alternative technology products **124** available for each of the technology components **122***a*-**122***n*. The alternative technology products **124** identified for each technology component **122***a*-**122***n* may include the current technology product **124***a*-**124***n* being used in the current application environment **120** for the technology components **122***a*-**122***n*. Application manager **130** may build a plurality of model application environments **136** for the software application **112** by selecting for each model application environment **136** a different combination of the technology products **124** for the technology components **122***a*-**122***n*. That is, each model application environment **136** is built to use a combination of technology products **124** that is different from the combinations of technology products **124** used for other model application environments **136** built for the software application **112**. In one embodiment, the model application environments **136** include a copy of the current application environment **120**. Application manager **130** builds the copy of the current application environment **120** as part of building the model application environments **136** for comparing with other model application environments **136**.

[0041] In one example, when the technology components **122***a*-**122***n* of the current application environment **120** includes a middleware component, a database engine component and a virtualization technique component, application manager **130** may identify alternative technology products **124** for each of a middleware component, a database engine component and a virtualization technique component. Application manager **130** may build a plurality of potential model application environments **136** for the software application **112** by using different combinations of technology products **124** for the middleware component, database engine component and virtualization technique component. For example, application manager **130** may build each model application environment by selecting a different alternative technology product **124** for at least one of the technology components **122***a*-**122***n*. Additionally, application manager **130** may build a copy of the current application environment **120** by selecting technology products **124***a*-**124***n* for the technology components **122***a*-**122***n*.

[0042] Application manager **130** may automatically configure each of the model application environments **136** to suit the software application **112**. The application manager **130** may obtain information relating to how the software application **112** is configured with respect to one or more parameters (e.g., logging-in, load balancing, database details etc.) in the current application environment **120**. Application manager **130** may configure each of the model application environments **136** for the software application **112**, based on the configurations in the current application environment **120**, for example, to match the configurations in the current application environment **120**. Application manager **130** may also configure one or more parameters of a model application environment by modifying the configuration of the parameters in the current application environment **120** to suit the model application environment **136**.

[0043] Application manager **130** may maintain information relating to performance benchmarks **142** for each technology product **124** under distinct performance conditions such as user traffic and amount of data being handled. A performance benchmark **142** relating to a technology product **124** includes a level of performance provided by the technology product **124** corresponding to at least one performance metric **138**. A performance metric **138**, as discussed below, may relate to a load performance and/or processing speed performance of the technology product **124**. For example, a technology product **124** may be asso-

ciated with high load performance meaning the technology product **124** can handle a large amount of user traffic and/or large amount of data. Another technology product **124** may be associated with high server performance which may mean that the technology product **124** provides fast server response times. The performance benchmarks **142** of the technology products **124** may allow the application manager **130** to build custom model application environments by selecting one or more technology products **124** that satisfy specific performance requirements of the software application **112**. For example, application manager **130** may obtain information relating to at least one performance requirement of the software application **112** corresponding to a technology component **122**. For example, application manager **130** may receive a request for a level of performance associated with the technology component **122**. Application manager **130** may determine a technology product **124** available for the technology component **122** and that satisfies the performance requirement, based on the performance benchmark **142** stored for the technology product **124**. Application manager **130** may build a model application environment **136** for the software application **112** by selecting the technology product **124** for the technology component **122**.

[0044] Application manager **130** may be configured to evolve to accommodate new and emerging technologies. For example, when a new technology product **124** is added for a technology component **122**, application manager **130** may assess whether the newly added technology product **124** can be used in an application environment for the software application **112**. For example, when the newly added technology product **124** corresponds to a technology component **122***a*-**122***n* being used by the current technology environment **120** of the software application **112**, application manager **130** may determine that the newly added technology product **124** may be used in the application environment of the software application **112**. Additionally or alternatively, application manager **130** may determine whether the newly added technology product **124** satisfies one or more performance requirements of the software application **112**. Application manager **130** may build a new model application environment **136** by selecting the newly added technology product **124** for the respective technology component **122**. Application manager **130** may build the new model application environment **136** in response to determining that the newly added technology product **124** can be used in the application environment of the software application **112** and/or that the newly added technology product **124** satisfies at least one requirements of the software application **112**. In one embodiment, application manager **130** may collect information relating to one or more performance requirements of the software application **112** according to a predetermined schedule (e.g., periodically). Application manager **130** may detect whether the performance requirements of the software application **112** has changed and build a customized application environment for the software application **112** based on current performance requirements of the software application **112**. This may ensure that the current application environment **120** of the software application **112** evolves to accommodate the changing performance requirements of the software application **112**.

[0045] At step **206**, application manager **130** runs the software application **112** using each of the model application environments (including the copy of the current application environment **120**) in a simulated environment. Application manager **130** may include a simulator logic that can be configured to host and deploy each of the model application environments **136** (including the current application environment **120**) and run the software application **112** in each of the model application environments **136**.

[0046] At step **208**, application manager **130** collects a plurality of performance metrics **138** related to performance of the software application **112** in the current application environment **120** and each of the model application environments **136** while running in the simulated environment. In one embodiment, application manager **130** runs each model application environment **136** in the simulated environment and collects performance metrics **138** for the model application environment **136** before moving on to the next model application environment.

[0047] Application manager **130** may collect data relating to a plurality of performance metrics **138** related to performance of the software application in each of the model application environments **136** (including the current application environment **120**) while running in the simulated environment. The plurality of performance metrics **138** may include, but are not limited to, one or more of application server performance, performance related to type of data handled by the software application, database performance, data security analysis, input and output analysis and deployment on cloud platforms. Application server performance may relate to response time of the application server at different levels of load (e.g., user traffic, amount of data etc.). Performance data related to the type of data handled by the software application **112** may include information relating to whether the software application **112** handles sensitive data needing a higher level of security and whether a model application environment **136** can handle sensitive data. This performance metric may allow the application manager **130** to recommend secure technology products **124** that can securely handle sensitive user data, such as private cloud product or a virtual machine where the virtualization is done at the hardware level to provide maximum security. Database performance may relate to database response times. Data relating to data security analysis may include security vulnerabilities (e.g., relating to data security) of the software application **112** in particular model application environments **136**. Application manager **130** scans the software application **112** for known vulnerabilities in each model application environment **136** and for potential threats to the application **112**. Application manager **130** may determine based on this performance metric whether a particular model application environment **136** is vulnerable to attacks. Data related to deployment on cloud platforms may include information relating to whether the software application **112** can be deployed on cloud platforms. For example, application manager **130** may analyze whether the software application **112** is suitable for deploying on a cloud platform based on running the software application **112** in model application environments **136** including cloud deployments. This performance metric may help the application manager **130** determine whether the software application **112** can be deployed on cloud platforms or is more suitable for a private in-house server or virtual machine deployment.

[0048] To measure the performance of the software application **112** in each model application environment **136**, application manager **130** may run one or more tests on each model application environment **136** while the simulator logic is running the software application **112** in the model

application environment **136**. Application manager **130** may collect one or more of the performance metrics **138** described above corresponding to each model application environment **136** as a result of conducting the one or more tests. The tests may include load testing to test the software application **112** at different amounts of load (e.g., user traffic, amount of data etc.). For example, load testing may include simulating simultaneous logins of different numbers of users using different amounts of test data. Additionally or alternatively, the tests may include performance testing to test the overall performance of a model application environment **136** and collect data on performance metrics **142** including availability, response time and stability of the software application **112**. In one example, for each model application environment **136**, application manager **130** carries out the testing in multiple cycles with different amounts of load. For example, each model application environment **136** is tested initially with 10 k simultaneous user logins and 100 GB of test data. This test cycle is then repeated three times with a replication load factor of 2×. This means with each test cycle, the load applied is two times that applied in the previous test cycle. In one embodiment, the load size may be customized based on the requirements of the software application **112**. In an additional embodiment, application manager **130** tests each model application environment **136** for security vulnerabilities (e.g., data security) and collects data relating to how vulnerable the model application environment **136** is to potential security threats.

[0049] At step, **210**, application manager **130** checks whether performance metrics **138** have been collected for all the model application environments **136** (including the current application environment **120**). If not, method **200** proceeds to step **212**, where application manager selects the next model application environment **136** for running the software application **112** in the simulated environment at step **206** and then collects performance metrics **138** for the next model application environment **136**. If application manager **130** determines that performance metrics **138** have been collected for all model application environments **136** including the current application environment **120**, method **200** proceeds to step **214**.

[0050] At step **214**, application manager **130** generates a recommendation report **140** based on the collected performance metrics **138**, wherein the recommendation report **140** includes a recommendation of a different technology product **124** for at least one of the technology components **122***a*-**122***n* used in the current application environment **120**, wherein the different technology product **124** is different from a current technology product **124***a*-**124***n* used for the at least one technology component **122***a*-**122***n* in the current application environment **120**.

[0051] Application manager **130** may analyze the performance metrics **138** collected for each model application environment **136** (including the current application environment **120**) and generate the recommendation report **140** based on the analysis. The recommendation report **140** may include a recommendation to use one or more model application environments **136** for the software application **112** that are found to perform better than the current application environment **120** based on the analysis of the performance metrics **138**. Additionally or alternatively, the recommendation report **140** may include recommendations to use individual technology products **124** for one or more technology components **122** that satisfy one or more performance requirements of the software application **112**.

[0052] In order to analyze the performance metrics **138**, application manager **130** may segregate the performance metrics **138** collected for each model application environment **136** including the current application environment **120**. The segregating may include grouping together data collected for each performance metric **138** for all the model application environments **136**. For example, data relating to application server performance metric collected for all model application environments **136** may be grouped together. Similarly, data relating to database performance metric collected for all model application environments **136** may be grouped together. Application manager **130** may compare the model application environments **136** based on each performance metric **138** collected for the respective model application environments **136**. For example, the application server performance metric of all model application environments **136** may be compared separately. Similarly, the database performance metric of all model application environments **136** may be compared separately. Application manager **130** may recommend one or more model application environments **136** and/or individual technology products **124** based on results of the comparison.

[0053] In one or more embodiments, for each model application environment **136** (including the current application environment **120**), application manager **130** assigns a performance score to each performance metric **138** collected for the model application environment **136**. Application manager **130** may assign a higher score for a higher level of performance related to a performance metric **138**. For example, if a first model application environment **136** is found to be able to handle **500** simultaneous user logins at one time and a second model application environment **136** is found to be able to handle **10***k* simultaneous user logins at one time, the application performance metric for the second model application environment **136** is assigned a higher performance score than the application performance metric for the first model application environment **136**. Application manager **130** may compare the respective scores of each performance metric **138** collected for several model application environments **136**. Application manager **130** may recommend a model application environment **136** that is found to have a higher score with regard to one or more performance metrics **138** as compared to the respective scores of the one or more performance metrics **138** collected for other model application environments **136**. Following the above example, application manager **130** may recommend the second model application environment **136** for the software application **112** over the first model application environment **136** based on the higher score associated with the application server performance metric collected for the second model application environment **136**.

[0054] In one or more embodiments, application manager **130** may compare the total performance scores of multiple performance metrics **138** collected for each model application environment **136** and recommend a model application environment **136** that has the highest total performance score for the performance metrics **138**. For example, application manager **130** may compare the total performance scores of all performance metrics **138** collected for each model application environment **136** and recommend a model application environment **136** that has the highest total performance score of for the performance metrics **138**.

[0055] In one or more embodiments, application manager **130** may sort the performance metrics **138** collected for each model application environment **136** (including the current application environment **120**) based on one or more requirements of the software application **112**, for example, by placing a higher priority on performance metrics **138** that are associated with the one or more requirements of the software application **112**. Application manager **130** may recommend one or more model application environments **136** and or individual technology products **124** that have higher scores recorded for the prioritized performance metrics **138** associated with the one or more requirements of the software application **112**, regardless of scores recorded for other performance metrics **138**. For example, the software application **112** may require high database performance (e.g., quick response times) and high data security as a result of handling sensitive user data. Accordingly, application manager **130** may prioritize the database performance metric and the security performance metric over other performance metrics **138** collected for the model application environments **136**. In this case, application manager **130** recommends one or more model application environments **136** that were found to have the highest scores for the database performance metric and the security performance metric. Application manager **130** may obtain the one or more requirements of the software application **112** in response to receiving a request for the one or more requirements or by automatically determining the one or more requirements by analyzing one or more performance metrics **138** of the software application **112**. For example, based on analyzing the software application **112** while running using one or more model application environments **136**, application manager may determine that the software application is vulnerable to cyber attacks. This detection may prompt the application manager **130** to prioritize the data security metric for recommending model application environments **136** that were found to have the highest scores related to the data security metric.

[0056] In one embodiment, application manager **130** may compare one or more model application environments **136** with the current application environment **120** based on the performance scores of a prioritized performance metric **138** (e.g., database performance, application server performance etc.). If application manager **130** determines that a model application environment **136** has a higher score for the performance metric **138** as compared to the respective score of the performance metric **138** collected for the current application environment **120**, application manager **130** may determine at least one technology product **124** used for a corresponding technology component **122** associated with the performance metric **138** in the determined model application environment **136** that is different from the current technology product **124** used for the technology component **122** in the current application environment **120**. Application manager **130** recommend using the determined at least one technology product **124** for the technology component **122** to achieve a higher level of performance related to the prioritized performance metric **138**.

[0057] In an additional or alternative embodiment, application manager **130** may receive a request for a higher level of performance related to a performance metric **138** associated with at least one technology component **122** as compared to a current level of performance related to the performance metric **138** in the current application environment **120**. In response, application manager **130** may compare one or more model application environments **136** with the current application environment **120** based on the performance scores of the requested performance metric **138**. Based on the results of the comparison, application manager **130** may determine a model application environment **136** that yielded a higher score for the requested performance metric **138** compared to the score of the performance metric **138** in the current application environment **120**. Application manager **130** may determine a technology product **124** used for the at least one technology component **122** in the determined model application environment **136** and recommends using the determined technology product **124** for the at least one technology component **122** to achieve the requested higher level of performance related to the performance metric **138**.

[0058] FIG. **3** is a flowchart of an example method **300** for building application environments for a software application, in accordance with certain embodiments of the present disclosure. Method **300** may be performed by application manager **130** as shown in FIG. **1** and described above.

[0059] At step **302**, application manager **130** obtains information relating to a current application environment **120** being used to run a software application **112**. As described above, current application system **110** may run the software application **112** in the current application environment **120**. The current application environment **120** may use technology components (shown as **122***a*-**122***n*), wherein each technology component **122***a*-**122***n* uses a respective technology product (shown as **124***a*-**124***n*) of a plurality of alternative technology products available for the technology component **122***a*-**122***n*. The technology components **122***a*-**122***n* may include one or more hardware technology components, one or more software technology components or a combination there of. For example, each technology component **122***a*-**122***n* may refer to a type of hardware or software tool used to implement, run and/or support the software application **112**. For example, a technology component may include, but is not limited to, a middleware component, an operating system component, a database engine component, an implementation technology component (e.g., coding language used to implement the software application **112**), a cloud provider component, a network technology component (e.g., WAN, LAN, VPN etc.), a virtualization technique component and a server less computing technology component. A technology product **124***a*-**124***n* may refer to a specific hardware or software tool/technology used for a corresponding technology component. For example, Apache Tomcat® can be a technology product used as a middleware component. Thus, the term "technology component" may generally refer to a type of technology product, and the term "technology product" may refer to a specific hardware or software tool used for a respective technology component. In one example case, current application environment **120** may include three technology components **122***a*, **122***b* and **122***c* using respective technology products **124***a*, **124***b* and **124***c*. Technology component **122***a* may be a middleware component that uses Apache Tomcat server as the specific technology product **124***a*. Technology component **122***b* may be a database engine component that uses SQL server as the specific technology product **124***b*. Technology component **122***c* may be a virtualization technique component that uses VMware virtual machine as the specific technology product **124***c*.

[0060] Application manager **130** may obtain information relating to the current application environment **120** being used for the software application **112**. In one embodiment, application manager **130** may use an application connector to obtain the information relating to the current application environment **120**. The application connector may be a software agent designed to run on the current application system **110** and extract information relating to the current application environment **120** from the current application system **110**. The information extracted by the application connector may include information relating to the particular technology components **122***a*-**122***n* of the current application environment **120** and the specific technology products **124***a*-**124***n* being used for each technology component **122***a*-**122***n*. In one embodiment, the information extracted by the application connecter may include information relating to one or more performance requirements of the software application with regard to one or more performance metrics **138** including load performance and processing performance. The information relating to the performance requirements of the software application may be used by application manager **130** to recommend one or more model application environments **136** and/or technology products **124** that suit the performance requirements.

[0061] At step **304**, based on the obtained information related to the current application environment **120**, application manager **130** identifies the one or more technology components **122***a*-**122***n* being used in the current application environment **120** to run the software application **112**.

[0062] At step **306**, based on the obtained information related to the current application environment **120**, application manager **130** further identifies a plurality of alternative technology products available for each of the one or more technology components **120***a*-**120***n* used in the current application environment **120**.

[0063] As described above, several alternative technology products **124** are typically available corresponding to each technology component **122***a*-**122***n*. For example, technology product alternatives for the operating system (OS) component may include, but are not limited to, Windows OS, Linux OS, Redhat OS and Fedora OS. Technology product alternatives for the implementation technology component may include, but are not limited to, C++, Java, C#, Python and JavaScript. Technology product alternatives for the middleware technology component may include, but are not limited to, Apache Tomcat server, JBoss server and Oracle server. Technology product alternatives for the cloud component may include, but are not limited to, Microsoft Azure, Amazon webservices and Google cloud platform. Technology product alternatives for the network component may include, but are not limited to, Local Area Network (LAN), Wireless (WLAN), Wide Area Network (WAN) and Virtual Private Network (VPN). Technology product alternatives for the virtualization technique component may include, but are not limited to, VMware, Microsoft Hyper-V, Openstack and Xen.

[0064] Based on the information obtained from the current application system **110**, application manager **130** may build a plurality of alternative model application environments **136** that can potentially be used for the software application **112**. For example, based on the information obtained from the current application system **110**, application manager **130** may determine the technology components **122***a*-**122***n* that need to be incorporated into an application environment meant for the software application **112**. Once, the technology components **122***a*-**122***n* are identified, application manager **130** may identify the plurality of alternative technology products **124** available for each of the technology components **122***a*-**122***n*. The alternative technology products **124** identified for each technology component **122***a*-**122***n* may include the current technology product **124***a*-**124***n* being used in the current application environment **120** for the technology components **122***a*-**122***n*. Application manager **130** may build a plurality of model application environments **136** for the software application **112** by selecting for each model application environment **136** a different combination of the technology products **124** for the technology components **122***a*-**122***n*. That is, each model application environment **136** is built to use a combination of technology products **124** that is different from the combinations of technology products **124** used for other model application environments **136** built for the software application **112**. In one embodiment, the model application environments **136** include a copy of the current application environment **120**. Application manager **130** builds the copy of the current application environment **120** as part of building the model application environments **136** for comparing with other model application environments **136**.

[0065] In one example, when the technology components **122***a*-**122***n* of the current application environment **120** includes a middleware component, a database engine component and a virtualization technique component, application manager **130** may identify alternative technology products **124** for each of a middleware component, a database engine component and a virtualization technique component. Application manager **130** may build a plurality of potential model application environments **136** for the software application **112** by using different combinations of technology products **124** for the middleware component, database engine component and virtualization technique component. For example, application manager **130** may build each model application environment by selecting a different alternative technology product **124** for at least one of the technology components **122***a*-**122***n*. Additionally, application manager **130** may build a copy of the current application environment **120** by selecting technology products **124***a*-**124***n* for the technology components **122***a*-**122***n*.

[0066] Application manager **130** may automatically configure each of the model application environments **136** to suit the software application **112**. The application manager **130** may obtain information relating to how the software application **112** is configured with respect to one or more parameters (e.g., logging-in, load balancing, database details etc.) in the current application environment **120**. Application manager **130** may configure each of the model application environments **136** for the software application **112**, based on the configurations in the current application environment **120**, for example, to match the configurations in the current application environment **120**. Application manager **130** may also configure one or more parameters of a model application environment by modifying the configuration of the parameters in the current application environment **120** to suit the model application environment **136**.

[0067] At step **308**, application manager **130** receives a request for a level of performance associated with a technology component **120***a*-**120***n*.

[0068] Application manager **130** may maintain information relating to performance benchmarks **142** for each tech-

nology product **124** under distinct performance conditions such as user traffic and amount of data being handled. A performance benchmark **142** relating to a technology product **124** includes a level of performance provided by the technology product **124** corresponding to at least one performance metric **138**. A performance metric **138**, as discussed below, may relate to a load performance and/or processing speed performance of the technology product **124**. For example, a technology product **124** may be associated with high load performance meaning the technology product **124** can handle a large amount of user traffic and/or large amount of data. Another technology product **124** may be associated with high server performance which may mean that the technology product **124** provides fast server response times. The performance benchmarks **142** of the technology products **124** may allow the application manager **130** to build custom model application environments by selecting one or more technology products **124** that satisfy specific performance requirements of the software application **112**. For example, application manager **130** may obtain information relating to at least one performance requirement of the software application **112** corresponding to a technology component **122**. For example, application manager **130** may receive a request for a level of performance associated with the technology component **122**.

[0069] At step **310**, Application manager **130** may determine whether a technology product **124** is available for the technology component **122** that satisfies the performance requirement associated with the requested level of performance, based on performance benchmarks **142** stored for the alternative technology products **124** available for the technology component **122**. If a suitable technology product **124** is not available, method **300** ends here. However, if application manager **130** determines a technology product **124** available for the technology component **122** that satisfies the requested level of performance, method **300** proceeds to step **312**, where application manager **130** selects the determined technology product **124**.

[0070] At step **314**, application manager **130** builds a model application environment **136** for the software application **112** by selecting the technology product **124** for the technology component **122**.

[0071] In one or more embodiments, application manager **130** may be configured to evolve to accommodate new and emerging technologies. For example, when a new technology product **124** is added for a technology component **122**, application manager **130** may assess whether the newly added technology product **124** can be used in an application environment for the software application **112**. For example, when the newly added technology product **124** corresponds to a technology component **122***a*-**122***n* being used by the current technology environment **120** of the software application **112**, application manager **130** may determine that the newly added technology product **124** may be used in the application environment of the software application **112**. Additionally or alternatively, application manager **130** may determine whether the newly added technology product **124** satisfies one or more performance requirements of the software application **112**. Application manager **130** may build a new model application environment **136** by selecting the newly added technology product **124** for the respective technology component **122**. Application manager **130** may build the new model application environment **136** in response to determining that the newly added technology product **124** can be used in the application environment of the software application **112** and/or that the newly added technology product **124** satisfies at least one requirements of the software application **112**. In one embodiment, application manager **130** may collect information relating to one or more performance requirements of the software application **112** according to a pre-determined schedule (e.g., periodically). Application manager **130** may detect whether the performance requirements of the software application **112** has changed and build a customized application environment for the software application **112** based on current performance requirements of the software application **112**. This may ensure that the current application environment **120** of the software application **112** evolves to accommodate the changing performance requirements of the software application **112**.

[0072] FIG. **4** is a flowchart of an example method **400** for improving an application environment for a software application, in accordance with certain embodiments of the present disclosure. Method **400** may be performed by application manager **130** as shown in FIG. **1** and described above.

[0073] At step **402**, application manager **130** obtains information relating to a current application environment **120** being used to run a software application **112**. As described above, current application system **110** may run the software application **112** in the current application environment **120**. The current application environment **120** may use technology components (shown as **122***a*-**122***n*), wherein each technology component **122***a*-**122***n* uses a respective technology product (shown as **124***a*-**124***n*). The technology components **122***a*-**122***n* may include one or more hardware technology components, one or more software technology components or a combination there of. For example, each technology component **122***a*-**122***n* may refer to a type of hardware or software tool used to implement, run and/or support the software application **112**. For example, a technology component may include, but is not limited to, a middleware component, an operating system component, a database engine component, an implementation technology component (e.g., coding language used to implement the software application **112**), a cloud provider component, a network technology component (e.g., WAN, LAN, VPN etc.), a virtualization technique component and a server less computing technology component. A technology product **124***a*-**124***n* may refer to a specific hardware or software tool/technology used for a corresponding technology component. For example, Apache Tomcat® can be a technology product used as a middleware component. Thus, the term "technology component" may generally refer to a type of technology product, and the term "technology product" may refer to a specific hardware or software tool used for a respective technology component. In one example case, current application environment **120** may include three technology components **122***a*, **122***b* and **122***c* using respective technology products **124***a*, **124***b* and **124***c*. Technology component **122***a* may be a middleware component that uses Apache Tomcat server as the specific technology product **124***a*. Technology component **122***b* may be a database engine component that uses SQL server as the specific technology product **124***b*. Technology component **122***c* may be a virtualization technique component that uses VMware virtual machine as the specific technology product **124***c*.

[0074] Application manager **130** may obtain information relating to the current application environment **120** being

used for the software application **112**. In one embodiment, application manager **130** may use an application connector to obtain the information relating to the current application environment **120**. The application connector may be a software agent designed to run on the current application system **110** and extract information relating to the current application environment **120** from the current application system **110**. The information extracted by the application connector may include information relating to the particular technology components **122***a*-**122***n* of the current application environment **120** and the specific technology products **124***a*-**124***n* being used for each technology component **122***a*-**122***n*. In one embodiment, the information extracted by the application connecter may include information relating to one or more performance requirements of the software application with regard to one or more performance metrics **138** including load performance and processing performance. As described below, the information relating to the performance requirements of the software application may be used by application manager **130** to recommend one or more model application environments **136** and/or technology products **124** that suit the performance requirements.

[0075] At step **404**, application manager **130** obtains information relating to a plurality of model application environments **136** built for the software application **112** based on the information obtained relating to the current application environment **120**.

[0076] As described above, several alternative technology products **124** are typically available corresponding to each technology component **122***a*-**122***n*. For example, technology product alternatives for the operating system (OS) component may include, but are not limited to, Windows OS, Linux OS, Redhat OS and Fedora OS. Technology product alternatives for the implementation technology component may include, but are not limited to, C++, Java, C#, Python and JavaScript. Technology product alternatives for the middleware technology component may include, but are not limited to, Apache Tomcat server, JBoss server and Oracle server. Technology product alternatives for the cloud component may include, but are not limited to, Microsoft Azure, Amazon webservices and Google cloud platform. Technology product alternatives for the network component may include, but are not limited to, Local Area Network (LAN), Wireless (WLAN), Wide Area Network (WAN) and Virtual Private Network (VPN). Technology product alternatives for the virtualization technique component may include, but are not limited to, VMware, Microsoft Hyper-V, Openstack and Xen.

[0077] As described above, the technology products **124***a*-**124***n* used for the respective technology components **122***a*-**122***n* in the current application environment **120** may not be the most optimal for the software application **112**. That is, the current application environment **120** may not always yield the performance desired for the software application **112**.

[0078] Based on the information obtained from the current application system **110**, application manager **130** may build a plurality of alternative model application environments **136** that can potentially be used for the software application **112**. For example, based on the information obtained from the current application system **110**, application manager **130** may determine the technology components **122***a*-**122***n* that need to be incorporated into an application environment meant for the software application **112**. Once, the technology components **122***a*-**122***n* are identified, application manager **130** may identify the plurality of alternative technology products **124** available for each of the technology components **122***a*-**122***n*. The alternative technology products **124** identified for each technology component **122***a*-**122***n* may include the current technology product **124***a*-**124***n* being used in the current application environment **120** for the technology components **122***a*-**122***n*. Application manager **130** may build a plurality of model application environments **136** for the software application **112** by selecting for each model application environment **136** a different combination of the technology products **124** for the technology components **122***a*-**122***n*. That is, each model application environment **136** is built to use a combination of technology products **124** that is different from the combinations of technology products **124** used for other model application environments **136** built for the software application **112**. In one embodiment, the model application environments **136** include a copy of the current application environment **120**. Application manager **130** builds the copy of the current application environment **120** as part of building the model application environments **136** for comparing with other model application environments **136**.

[0079] In one example, when the technology components **122***a*-**122***n* of the current application environment **120** includes a middleware component, a database engine component and a virtualization technique component, application manager **130** may identify alternative technology products **124** for each of a middleware component, a database engine component and a virtualization technique component. Application manager **130** may build a plurality of potential model application environments **136** for the software application **112** by using different combinations of technology products **124** for the middleware component, database engine component and virtualization technique component. For example, application manager **130** may build each model application environment by selecting a different alternative technology product **124** for at least one of the technology components **122***a*-**122***n*. Additionally, application manager **130** may build a copy of the current application environment **120** by selecting technology products **124***a*-**124***n* for the technology components **122***a*-**122***n*.

[0080] Application manager **130** may automatically configure each of the model application environments **136** to suit the software application **112**. The application manager **130** may obtain information relating to how the software application **112** is configured with respect to one or more parameters (e.g., logging-in, load balancing, database details etc.) in the current application environment **120**. Application manager **130** may configure each of the model application environments **136** for the software application **112**, based on the configurations in the current application environment **120**, for example, to match the configurations in the current application environment **120**. Application manager **130** may also configure one or more parameters of a model application environment by modifying the configuration of the parameters in the current application environment **120** to suit the model application environment **136**.

[0081] Application manager **130** may maintain information relating to performance benchmarks **142** for each technology product **124** under distinct performance conditions such as user traffic and amount of data being handled. A performance benchmark **142** relating to a technology product **124** includes a level of performance provided by the

technology product **124** corresponding to at least one performance metric **138**. A performance metric **138**, as discussed below, may relate to a load performance and/or processing speed performance of the technology product **124**. For example, a technology product **124** may be associated with high load performance meaning the technology product **124** can handle a large amount of user traffic and/or large amount of data. Another technology product **124** may be associated with high server performance which may mean that the technology product **124** provides fast server response times. The performance benchmarks **142** of the technology products **124** may allow the application manager **130** to build custom model application environments by selecting one or more technology products **124** that satisfy specific performance requirements of the software application **112**. For example, application manager **130** may obtain information relating to at least one performance requirement of the software application **112** corresponding to a technology component **122**. For example, application manager **130** may receive a request for a level of performance associated with the technology component **122**. Application manager **130** may determine a technology product **124** available for the technology component **122** and that satisfies the performance requirement, based on the performance benchmark **142** stored for the technology product **124**. Application manager **130** may build a model application environment **136** for the software application **112** by selecting the technology product **124** for the technology component **122**.

[0082] Application manager **130** may be configured to evolve to accommodate new and emerging technologies. For example, when a new technology product **124** is added for a technology component **122**, application manager **130** may assess whether the newly added technology product **124** can be used in an application environment for the software application **112**. For example, when the newly added technology product **124** corresponds to a technology component **122***a*-**122***n* being used by the current technology environment **120** of the software application **112**, application manager **130** may determine that the newly added technology product **124** may be used in the application environment of the software application **112**. Additionally or alternatively, application manager **130** may determine whether the newly added technology product **124** satisfies one or more performance requirements of the software application **112**. Application manager **130** may build a new model application environment **136** by selecting the newly added technology product **124** for the respective technology component **122**. Application manager **130** may build the new model application environment **136** in response to determining that the newly added technology product **124** can be used in the application environment of the software application **112** and/or that the newly added technology product **124** satisfies at least one requirements of the software application **112**. In one embodiment, application manager **130** may collect information relating to one or more performance requirements of the software application **112** according to a predetermined schedule (e.g., periodically). Application manager **130** may detect whether the performance requirements of the software application **112** has changed and build a customized application environment for the software application **112** based on current performance requirements of the software application **112**. This may ensure that the current application environment **120** of the software application **112** evolves to accommodate the changing performance requirements of the software application **112**.

[0083] At step **406**, application manager **130** runs the software application **112** using each of the model application environments (including the copy of the current application environment **120**) in a simulated environment. Application manager **130** may include a simulator logic that can be configured to host and deploy each of the model application environments **136** (including the current application environment **120**) and run the software application **112** in each of the model application environments **136**.

[0084] At step **408**, application manager **130** collects a plurality of performance metrics **138** related to performance of the software application **112** in the current application environment **120** and each of the model application environments **136** while running in the simulated environment. In one embodiment, application manager **130** runs each model application environment **136** in the simulated environment and collects performance metrics **138** for the model application environment **136** before moving on to the next model application environment.

[0085] Application manager **130** may collect data relating to a plurality of performance metrics **138** related to performance of the software application in each of the model application environments **136** (including the current application environment **120**) while running in the simulated environment. The plurality of performance metrics **138** may include, but are not limited to, one or more of application server performance, performance related to type of data handled by the software application, database performance, data security analysis, input and output analysis and deployment on cloud platforms. Application server performance may relate to response time of the application server at different levels of load (e.g., user traffic, amount of data etc.). Performance data related to the type of data handled by the software application **112** may include information relating to whether the software application **112** handles sensitive data needing a higher level of security and whether a model application environment **136** can handle sensitive data. This performance metric may allow the application manager **130** to recommend secure technology products **124** that can securely handle sensitive user data, such as private cloud product or a virtual machine where the virtualization is done at the hardware level to provide maximum security. Database performance may relate to database response times. Data relating to data security analysis may include security vulnerabilities (e.g., relating to data security) of the software application **112** in particular model application environments **136**. Application manager **130** scans the software application **112** for known vulnerabilities in each model application environment **136** and for potential threats to the application **112**. Application manager **130** may determine based on this performance metric whether a particular model application environment **136** is vulnerable to attacks. Data related to deployment on cloud platforms may include information relating to whether the software application **112** can be deployed on cloud platforms. For example, application manager **130** may analyze whether the software application **112** is suitable for deploying on a cloud platform based on running the software application **112** in model application environments **136** including cloud deployments. This performance metric may help the application manager **130** determine whether the software application **112** can be

deployed on cloud platforms or is more suitable for a private in-house server or virtual machine deployment.

[0086] To measure the performance of the software application **112** in each model application environment **136**, application manager **130** may run one or more tests on each model application environment **136** while the simulator logic is running the software application **112** in the model application environment **136**. Application manager **130** may collect one or more of the performance metrics **138** described above corresponding to each model application environment **136** as a result of conducting the one or more tests. The tests may include load testing to test the software application **112** at different amounts of load (e.g., user traffic, amount of data etc.). For example, load testing may include simulating simultaneous logins of different numbers of users using different amounts of test data. Additionally or alternatively, the tests may include performance testing to test the overall performance of a model application environment **136** and collect data on performance metrics **142** including availability, response time and stability of the software application **112**. In one example, for each model application environment **136**, application manager **130** carries out the testing in multiple cycles with different amounts of load. For example, each model application environment **136** is tested initially with 10 k simultaneous user logins and 100 GB of test data. This test cycle is then repeated three times with a replication load factor of 2×. This means with each test cycle, the load applied is two times that applied in the previous test cycle. In one embodiment, the load size may be customized based on the requirements of the software application **112**. In an additional embodiment, application manager **130** tests each model application environment **136** for security vulnerabilities (e.g., data security) and collects data relating to how vulnerable the model application environment **136** is to potential security threats.

[0087] At step, **410**, application manager **130** checks whether performance metrics **138** have been collected for all the model application environments **136** (including the current application environment **120**). If not, method **400** proceeds to step **412**, where application manager selects the next model application environment **136** for running the software application **112** in the simulated environment at step **206** and then collects performance metrics **138** for the next model application environment **136**. If application manager **130** determines that performance metrics **138** have been collected for all model application environments **136** including the current application environment **120**, method **400** proceeds to step **414**.

[0088] At step **414**, application manager **130** assigns a score to each of the performance metrics **138** collected for each of the model application environments including the current application environment **120**, wherein a higher score is assigned to a higher level of performance related to the performance metric **138**.

[0089] As described above, Application manager **130** may analyze the performance metrics **138** collected for each model application environment **136** (including the current application environment **120**) and generate a recommendation report **140** based on the analysis. The recommendation report **140** may include a recommendation to use one or more model application environments **136** for the software application **112** that are found to perform better than the current application environment **120** based on the analysis of the performance metrics **138**. Additionally or alternatively, the recommendation report **140** may include recommendations to use individual technology products **124** for one or more technology components **122** that satisfy one or more performance requirements of the software application **112**.

[0090] For each model application environment **136** (including the current application environment **120**), application manager **130** assigns a performance score to each performance metric **138** collected for the model application environment **136**. Application manager **130** may assign a higher score for a higher level of performance related to a performance metric **138**. For example, if a first model application environment **136** is found to be able to handle **500** simultaneous user logins at one time and a second model application environment **136** is found to be able to handle **10***k* simultaneous user logins at one time, the application performance metric for the second model application environment **136** is assigned a higher performance score than the application performance metric for the first model application environment **136**. Application manager **130** may compare the respective scores of each performance metric **138** collected for several model application environments **136**. Application manager **130** may recommend a model application environment **136** that is found to have a higher score with regard to one or more performance metrics **138** as compared to the respective scores of the one or more performance metrics **138** collected for other model application environments **136**. Following the above example, application manager **130** may recommend the second model application environment **136** for the software application **112** over the first model application environment **136** based on the higher score associated with the application server performance metric collected for the second model application environment **136**.

[0091] At step **416**, application manager **130** determines a model application environment **136** that yielded a higher score for a performance metric **138** compared to the score of the performance metric **138** in the current application environment. For example, application manager **130** may compare one or more model application environments **136** with the current application environment **120** based on the performance scores of a prioritized performance metric **138** (e.g., database performance, application server performance etc.).

[0092] At step **418**, when application manager **130** determines that the model application environment **136** has a higher score for the performance metric **138** as compared to the respective score of the performance metric **138** collected for the current application environment **120**, application manager **130** determines at least one technology product **124** used for a corresponding technology component **122** associated with the performance metric **138** in the determined model application environment **136** that is different from the current technology product **124** used for the technology component **122** in the current application environment **120**.

[0093] At step **420**, Application manager **130** recommends using the determined at least one technology product **124** for the technology component **122** in the application environment of the software application **112**, to achieve a higher level of performance related to the performance metric **138**.

[0094] In additional or alternative embodiments, application manager **130** may segregate the performance metrics **138** collected for each model application environment **136** including the current application environment **120**. The segregating may include grouping together data collected for

each performance metric **138** for all the model application environments **136**. For example, data relating to application server performance metric collected for all model application environments **136** may be grouped together. Similarly, data relating to database performance metric collected for all model application environments **136** may be grouped together. Application manager **130** may compare the model application environments **136** based on each performance metric **138** collected for the respective model application environments **136**, for example, based on performance scores of the performance metric **138** collected for the model application environments. For example, the application server performance metric of all model application environments **136** may be compared separately. Similarly, the database performance metric of all model application environments **136** may be compared separately. Application manager **130** may recommend one or more model application environments **136** and/or individual technology products **124** based on results of the comparison.

[0095] In one or more embodiments, application manager **130** may compare the total performance scores of multiple performance metrics **138** collected for each model application environment **136** and recommend a model application environment **136** that has the highest total performance score for the performance metrics **138**. For example, application manager **130** may compare the total performance scores of all performance metrics **138** collected for each model application environment **136** and recommend a model application environment **136** that has the highest total performance score of for the performance metrics **138**.

[0096] In one or more embodiments, application manager **130** may sort the performance metrics **138** collected for each model application environment **136** (including the current application environment **120**) based on one or more requirements of the software application **112**, for example, by placing a higher priority on performance metrics **138** that are associated with the one or more requirements of the software application **112**. Application manager **130** may recommend one or more model application environments **136** and or individual technology products **124** that have higher scores recorded for the prioritized performance metrics **138** associated with the one or more requirements of the software application **112**, regardless of scores recorded for other performance metrics **138**. For example, the software application **112** may require high database performance (e.g., quick response times) and high data security as a result of handling sensitive user data. Accordingly, application manager **130** may prioritize the database performance metric and the security performance metric over other performance metrics **138** collected for the model application environments **136**. In this case, application manager **130** recommends one or more model application environments **136** that were found to have the highest scores for the database performance metric and the security performance metric. Application manager **130** may obtain the one or more requirements of the software application **112** in response to receiving a request for the one or more requirements or by automatically determining the one or more requirements by analyzing one or more performance metrics **138** of the software application **112**. For example, based on analyzing the software application **112** while running using one or more model application environments **136**, application manager may determine that the software application is vulnerable to cyber attacks. This detection may prompt the application manager **130** to prioritize the data security metric for recommending model application environments **136** that were found to have the highest scores related to the data security metric.

[0097] In an additional or alternative embodiment, application manager **130** may receive a request for a higher level of performance related to a performance metric **138** associated with at least one technology component **122** as compared to a current level of performance related to the performance metric **138** in the current application environment **120**. In response, application manager **130** may compare one or more model application environments **136** with the current application environment **120** based on the performance scores of the requested performance metric **138**. Based on the results of the comparison, application manager **130** may determine a model application environment **136** that yielded a higher score for the requested performance metric **138** compared to the score of the performance metric **138** in the current application environment **120**. Application manager **130** may determine a technology product **124** used for the at least one technology component **122** in the determined model application environment **136** and recommends using the determined technology product **124** for the at least one technology component **122** to achieve the requested higher level of performance related to the performance metric **138**.

[0098] FIG. **5** is a flowchart of an example method **500** for analyzing performance metrics collected for model application environments of a software application, in accordance with certain embodiments of the present disclosure. Method **500** may be performed by application manager **130** as shown in FIG. **1** and described above.

[0099] At step **502**, application manager **130** obtains information about a plurality of performance metrics **138** related to performance of a software application **112** in a current application environment **120** and each of a plurality of model application environments **136**.

[0100] As described above, current application system **110** may run the software application **112** in the current application environment **120**. The current application environment **120** may use technology components (shown as **122***a*-**122***n*), wherein each technology component **122***a*-**122***n* uses a respective technology product (shown as **124***a*-**124***n*). The technology components **122***a*-**122***n* may include one or more hardware technology components, one or more software technology components or a combination there of. For example, each technology component **122***a*-**122***n* may refer to a type of hardware or software tool used to implement, run and/or support the software application **112**. For example, a technology component may include, but is not limited to, a middleware component, an operating system component, a database engine component, an implementation technology component (e.g., coding language used to implement the software application **112**), a cloud provider component, a network technology component (e.g., WAN, LAN, VPN etc.), a virtualization technique component and a server less computing technology component. A technology product **124***a*-**124***n* may refer to a specific hardware or software tool/technology used for a corresponding technology component. For example, Apache Tomcat® can be a technology product used as a middleware component. Thus, the term "technology component" may generally refer to a type of technology product, and the term "technology product" may refer to a specific hardware or software tool used for a

respective technology component. In one example case, current application environment **120** may include three technology components **122***a*, **122***b* and **122***c* using respective technology products **124***a*, **124***b* and **124***c*. Technology component **122***a* may be a middleware component that uses Apache Tomcat server as the specific technology product **124***a*. Technology component **122***b* may be a database engine component that uses SQL server as the specific technology product **124***b*. Technology component **122***c* may be a virtualization technique component that uses VMware virtual machine as the specific technology product **124***c*.

[0101] Application manager **130** may obtain information relating to the current application environment **120** being used for the software application **112**. In one embodiment, application manager **130** may use an application connector to obtain the information relating to the current application environment **120**. The application connector may be a software agent designed to run on the current application system **110** and extract information relating to the current application environment **120** from the current application system **110**. The information extracted by the application connector may include information relating to the particular technology components **122***a*-**122***n* of the current application environment **120** and the specific technology products **124***a*-**124***n* being used for each technology component **122***a*-**122***n*. In one embodiment, the information extracted by the application connecter may include information relating to one or more performance requirements of the software application with regard to one or more performance metrics **138** including load performance and processing performance. As described below, the information relating to the performance requirements of the software application may be used by application manager **130** to recommend one or more model application environments **136** and/or technology products **124** that suit the performance requirements.

[0102] Several alternative technology products **124** are typically available corresponding to each technology component **122***a*-**122***n*. For example, technology product alternatives for the operating system (OS) component may include, but are not limited to, Windows OS, Linux OS, Redhat OS and Fedora OS. Technology product alternatives for the implementation technology component may include, but are not limited to, C++, Java, C#, Python and JavaScript. Technology product alternatives for the middleware technology component may include, but are not limited to, Apache Tomcat server, JBoss server and Oracle server. Technology product alternatives for the cloud component may include, but are not limited to, Microsoft Azure, Amazon webservices and Google cloud platform. Technology product alternatives for the network component may include, but are not limited to, Local Area Network (LAN), Wireless (WLAN), Wide Area Network (WAN) and Virtual Private Network (VPN). Technology product alternatives for the virtualization technique component may include, but are not limited to, VMware, Microsoft Hyper-V, Openstack and Xen.

[0103] Based on the information obtained from the current application system **110**, application manager **130** may build a plurality of alternative model application environments **136** that can potentially be used for the software application **112**. For example, based on the information obtained from the current application system **110**, application manager **130** may determine the technology components **122***a*-**122***n* that need to be incorporated into an application environment meant for the software application **112**. Once, the technology components **122***a*-**122***n* are identified, application manager **130** may identify the plurality of alternative technology products **124** available for each of the technology components **122***a*-**122***n*. The alternative technology products **124** identified for each technology component **122***a*-**122***n* may include the current technology product **124***a*-**124***n* being used in the current application environment **120** for the technology components **122***a*-**122***n*. Application manager **130** may build a plurality of model application environments **136** for the software application **112** by selecting for each model application environment **136** a different combination of the technology products **124** for the technology components **122***a*-**122***n*. That is, each model application environment **136** is built to use a combination of technology products **124** that is different from the combinations of technology products **124** used for other model application environments **136** built for the software application **112**. In one embodiment, the model application environments **136** include a copy of the current application environment **120**. Application manager **130** builds the copy of the current application environment **120** as part of building the model application environments **136** for comparing with other model application environments **136**.

[0104] Application manager **130** may run the software application **112** using each of the model application environments (including the copy of the current application environment **120**) in a simulated environment and collect the plurality of performance metrics **138** related to performance of the software application **112** in the current application environment **120** and each of the model application environments **136** while running in the simulated environment.

[0105] Application manager **130** may collect data relating to the plurality of performance metrics **138** related to performance of the software application in each of the model application environments **136** (including the current application environment **120**) while running in the simulated environment. The plurality of performance metrics **138** may include, but are not limited to, one or more of application server performance, performance related to type of data handled by the software application, database performance, data security analysis, input and output analysis and deployment on cloud platforms. Application server performance may relate to response time of the application server at different levels of load (e.g., user traffic, amount of data etc.). Performance data related to the type of data handled by the software application **112** may include information relating to whether the software application **112** handles sensitive data needing a higher level of security and whether a model application environment **136** can handle sensitive data. This performance metric may allow the application manager **130** to recommend secure technology products **124** that can securely handle sensitive user data, such as private cloud product or a virtual machine where the virtualization is done at the hardware level to provide maximum security. Database performance may relate to database response times. Data relating to data security analysis may include security vulnerabilities (e.g., relating to data security) of the software application **112** in particular model application environments **136**. Application manager **130** scans the software application **112** for known vulnerabilities in each model application environment **136** and for potential threats to the application **112**. Application manager **130** may determine based on this performance metric whether a particular model

application environment **136** is vulnerable to attacks. Data related to deployment on cloud platforms may include information relating to whether the software application **112** can be deployed on cloud platforms. For example, application manager **130** may analyze whether the software application **112** is suitable for deploying on a cloud platform based on running the software application **112** in model application environments **136** including cloud deployments. This performance metric may help the application manager **130** determine whether the software application **112** can be deployed on cloud platforms or is more suitable for a private in-house server or virtual machine deployment.

[0106] To measure the performance of the software application **112** in each model application environment **136**, application manager **130** may run one or more tests on each model application environment **136** while the simulator logic is running the software application **112** in the model application environment **136**. Application manager **130** may collect one or more of the performance metrics **138** described above corresponding to each model application environment **136** as a result of conducting the one or more tests. The tests may include load testing to test the software application **112** at different amounts of load (e.g., user traffic, amount of data etc.). For example, load testing may include simulating simultaneous logins of different numbers of users using different amounts of test data. Additionally or alternatively, the tests may include performance testing to test the overall performance of a model application environment **136** and collect data on performance metrics **142** including availability, response time and stability of the software application **112**. In one example, for each model application environment **136**, application manager **130** carries out the testing in multiple cycles with different amounts of load. For example, each model application environment **136** is tested initially with 10 k simultaneous user logins and 100 GB of test data. This test cycle is then repeated three times with a replication load factor of 2×. This means with each test cycle, the load applied is two times that applied in the previous test cycle. In one embodiment, the load size may be customized based on the requirements of the software application **112**. In an additional embodiment, application manager **130** tests each model application environment **136** for security vulnerabilities (e.g., data security) and collects data relating to how vulnerable the model application environment **136** is to potential security threats.

[0107] At step **504**, application manager **130** assigns a performance score to each of the performance metrics collected for the current application environment and each of the model application environments.

[0108] Application manager **130** may analyze the performance metrics **138** collected for each model application environment **136** (including the current application environment **120**) and generate the recommendation report **140** based on the analysis. The recommendation report **140** may include a recommendation to use one or more model application environments **136** for the software application **112** that are found to perform better than the current application environment **120** based on the analysis of the performance metrics **138**. Additionally or alternatively, the recommendation report **140** may include recommendations to use individual technology products **124** for one or more technology components **122** that satisfy one or more performance requirements of the software application **112**.

[0109] To analyze the performance metrics **138**, for each model application environment **136** (including the current application environment **120**), application manager **130** assigns a performance score to each performance metric **138** collected for the model application environment **136**. Application manager **130** may assign a higher score for a higher level of performance related to a performance metric **138**. For example, if a first model application environment **136** is found to be able to handle **500** simultaneous user logins at one time and a second model application environment **136** is found to be able to handle **10***k* simultaneous user logins at one time, the application performance metric for the second model application environment **136** is assigned a higher performance score than the application performance metric for the first model application environment **136**.

[0110] At step **506**, application manager **130** compares the respective scores assigned to each performance metric **138** collected for the current application environment **120** and each of the model application environments **136**.

[0111] At step **508**, application manager **130** checks if all performance metrics **138** collected for all model application environments have been compared. If not, application manager **130** continues comparing the performance metrics at step **506**. When all performance metrics **138** collected for all model application environments have been compared, method **500** proceeds to step **510**.

[0112] At step **510**, application manager **130** detects that at least one model application environment **136** has a higher score associated with at least one performance metric **138** as compared to the respective score of the at least one performance metric **138** collected for the current application environment **120**.

[0113] At step **512**, application manager recommends the model application environment **136** that is found to have a higher score with regard to the at least one performance metric **138** as compared to the respective score of the at least one performance metrics **138** collected for the current application environment **120**. For example, application manager **130** may recommend a model application environment **136** for the software application **112** over the current application environment **120** based on a higher score associated with the application server performance metric collected for the model application environment **136**.

[0114] In additional or alternative embodiments, application manager **130** may segregate the performance metrics **138** collected for each model application environment **136** including the current application environment **120**. The segregating may include grouping together data collected for each performance metric **138** for all the model application environments **136**. For example, data relating to application server performance metric collected for all model application environments **136** may be grouped together. Similarly, data relating to database performance metric collected for all model application environments **136** may be grouped together. Application manager **130** may compare the model application environments **136** based on each performance metric **138** collected for the respective model application environments **136**, for example, based on performance scores of the performance metric **138** collected for the model application environments. For example, the application server performance metric of all model application environments **136** may be compared separately. Similarly, the database performance metric of all model application environments **136** may be compared separately. Application

manager **130** may recommend one or more model application environments **136** and/or individual technology products **124** based on results of the comparison.

[0115] In one or more embodiments, application manager **130** may compare the total performance scores of multiple performance metrics **138** collected for each model application environment **136** and recommend a model application environment **136** that has the highest total performance score for the performance metrics **138**. For example, application manager **130** may compare the total performance scores of all performance metrics **138** collected for each model application environment **136** and recommend a model application environment **136** that has the highest total performance score of for the performance metrics **138**.

[0116] In one or more embodiments, application manager **130** may sort the performance metrics **138** collected for each model application environment **136** (including the current application environment **120**) based on one or more requirements of the software application **112**, for example, by placing a higher priority on performance metrics **138** that are associated with the one or more requirements of the software application **112**. Application manager **130** may recommend one or more model application environments **136** and or individual technology products **124** that have higher scores recorded for the prioritized performance metrics **138** associated with the one or more requirements of the software application **112**, regardless of scores recorded for other performance metrics **138**. For example, the software application **112** may require high database performance (e.g., quick response times) and high data security as a result of handling sensitive user data. Accordingly, application manager **130** may prioritize the database performance metric and the security performance metric over other performance metrics **138** collected for the model application environments **136**. In this case, application manager **130** recommends one or more model application environments **136** that were found to have the highest scores for the database performance metric and the security performance metric. Application manager **130** may obtain the one or more requirements of the software application **112** in response to receiving a request for the one or more requirements or by automatically determining the one or more requirements by analyzing one or more performance metrics **138** of the software application **112**. For example, based on analyzing the software application **112** while running using one or more model application environments **136**, application manager may determine that the software application is vulnerable to cyber attacks. This detection may prompt the application manager **130** to prioritize the data security metric for recommending model application environments **136** that were found to have the highest scores related to the data security metric.

[0117] In one embodiment, application manager **130** may compare one or more model application environments **136** with the current application environment **120** based on the performance scores of a prioritized performance metric **138** (e.g., database performance, application server performance etc.). If application manager **130** determines that a model application environment **136** has a higher score for the performance metric **138** as compared to the respective score of the performance metric **138** collected for the current application environment **120**, application manager **130** may determine at least one technology product **124** used for a corresponding technology component **122** associated with the performance metric **138** in the determined model application environment **136** that is different from the current technology product **124** used for the technology component **122** in the current application environment **120**. Application manager **130** recommend using the determined at least one technology product **124** for the technology component **122** to achieve a higher level of performance related to the prioritized performance metric **138**.

[0118] In an additional or alternative embodiment, application manager **130** may receive a request for a higher level of performance related to a performance metric **138** associated with at least one technology component **122** as compared to a current level of performance related to the performance metric **138** in the current application environment **120**. In response, application manager **130** may compare one or more model application environments **136** with the current application environment **120** based on the performance scores of the requested performance metric **138**. Based on the results of the comparison, application manager **130** may determine a model application environment **136** that yielded a higher score for the requested performance metric **138** compared to the score of the performance metric **138** in the current application environment **120**. Application manager **130** may determine a technology product **124** used for the at least one technology component **122** in the determined model application environment **136** and recommends using the determined technology product **124** for the at least one technology component **122** to achieve the requested higher level of performance related to the performance metric **138**.

[0119] FIG. **6** illustrates an example schematic diagram of the application manager **130** illustrated in FIG. **1**, in accordance with one or more embodiments of the present disclosure.

[0120] Application manager **130** includes a processor **602**, a memory **606**, and a network interface **604**. The application manager **130** may be configured as shown in FIG. **6** or in any other suitable configuration.

[0121] The processor **602** comprises one or more processors operably coupled to the memory **606**. The processor **602** is any electronic circuitry including, but not limited to, state machines, one or more central processing unit (CPU) chips, logic units, cores (e.g. a multi-core processor), field-programmable gate array (FPGAs), application specific integrated circuits (ASICs), or digital signal processors (DSPs). The processor **602** may be a programmable logic device, a microcontroller, a microprocessor, or any suitable combination of the preceding. The processor **602** is communicatively coupled to and in signal communication with the memory **606**. The one or more processors are configured to process data and may be implemented in hardware or software. For example, the processor **602** may be 8-bit, 16-bit, 32-bit, 64-bit or of any other suitable architecture. The processor **602** may include an arithmetic logic unit (ALU) for performing arithmetic and logic operations, processor registers that supply operands to the ALU and store the results of ALU operations, and a control unit that fetches instructions from memory and executes them by directing the coordinated operations of the ALU, registers and other components.

[0122] The one or more processors are configured to implement various instructions. For example, the one or more processors are configured to execute instructions (e.g., application manager instructions **608**) to implement the

application manager **130**. In this way, processor **602** may be a special-purpose computer designed to implement the functions disclosed herein. In one or more embodiments, the application manager **130** is implemented using logic units, FPGAs, ASICs, DSPs, or any other suitable hardware. The application manager **130** is configured to operate as described with reference to FIGS. **1-5**. For example, the application manager **130** may be configured to perform at least a portion of the methods **200**, **300**, **400** and **500** as described in FIGS. **2**, **3**, **4** and **5** respectively.

[0123] The memory **606** comprises one or more disks, tape drives, or solid-state drives, and may be used as an over-flow data storage device, to store programs when such programs are selected for execution, and to store instructions and data that are read during program execution. The memory **606** may be volatile or non-volatile and may comprise a read-only memory (ROM), random-access memory (RAM), ternary content-addressable memory (TCAM), dynamic random-access memory (DRAM), and static random-access memory (SRAM).

[0124] The memory **606** is operable to store information relating to technology components **122**, technology products **124**, model application environments **136**, performance metrics **138**, recommendation reports **140**, performance benchmarks **142** and application manager instructions **608**. The application manager instructions **608** may include any suitable set of instructions, logic, rules, or code operable to execute the application manager **130**.

[0125] The network interface **604** is configured to enable wired and/or wireless communications. The network interface **604** is configured to communicate data between the application manager **130** and other devices, systems, or domains (e.g. current application system **110**). For example, the network interface **604** may comprise a Wi-Fi interface, a LAN interface, a WAN interface, a modem, a switch, or a router. The processor **602** is configured to send and receive data using the network interface **604**. The network interface **604** may be configured to use any suitable type of communication protocol as would be appreciated by one of ordinary skill in the art.

[0126] It may be noted that current application system **110** may be implemented similar to the application manager **130**. For example, the current application system **110** may include a processor and a memory storing instructions to implement the respective functionality as described above when executed by the processor. The memory of current application system **110** may store the software application **112** and the current application environment **120**.

[0127] While several embodiments have been provided in the present disclosure, it should be understood that the disclosed systems and methods might be embodied in many other specific forms without departing from the spirit or scope of the present disclosure. The present examples are to be considered as illustrative and not restrictive, and the intention is not to be limited to the details given herein. For example, the various elements or components may be combined or integrated in another system or certain features may be omitted, or not implemented.

[0128] In addition, techniques, systems, subsystems, and methods described and illustrated in the various embodiments as discrete or separate may be combined or integrated with other systems, modules, techniques, or methods without departing from the scope of the present disclosure. Other items shown or discussed as coupled or directly coupled or communicating with each other may be indirectly coupled or communicating through some interface, device, or intermediate component whether electrically, mechanically, or otherwise. Other examples of changes, substitutions, and alterations are ascertainable by one skilled in the art and could be made without departing from the spirit and scope disclosed herein.

[0129] To aid the Patent Office, and any readers of any patent issued on this application in interpreting the claims appended hereto, applicants note that they do not intend any of the appended claims to invoke 35 U.S.C. § 112(f) as it exists on the date of filing hereof unless the words "means for" or "step for" are explicitly used in the particular claim.

1. A system comprising:

at least one processor configured to:

obtain information relating to a current application environment being used to run a software application, wherein the current application environment comprises one or more technology components including one or more hardware technology components, one or more software technology components or a combination thereof, wherein each of the technology components uses a technology product from a plurality of alternative technology products available for the technology component;

build, based on the obtained information, a plurality of model application environments for the software application, wherein each of the model application environments comprises a different combination of the technology products used for the technology components;

run the software application using the current application environment and each of the model application environments in a simulated environment;

collect a plurality of performance metrics related to performance of the software application in the current application environment and each of the model application environments while running in the simulated environment; and

generate a recommendation report based on the performance metrics, wherein the recommendation report comprises a recommendation of a different technology product for at least one of the technology components used in the current application environment, wherein the different technology product is different from a current technology product used for the at least one technology component in the current application environment; and

a memory coupled to the at least one processor and configured to store information relating to the plurality of alternative technology products available for each of the technology components.

2. The system of claim **1**, wherein the information relating to the current application environment comprises information relating to the one or more technology components of the current application environment and the current technology product used for each of the one or more technology components.

3. The system of claim **2**, wherein the at least one processor is further configured to:

identify the plurality of alternative technology products for each of the one or more technology components, wherein each of the plurality of alternative technology

products is different from the current technology product of the technology component; and

build each of the plurality of model application environments by selecting for at least one of the technology components a technology product from the plurality of alternative technology products such that the model application environment uses a different combination of the technology products that is different from combinations of the technology products selected for other model application environments.

4. The system of claim **1**, wherein the memory is further configured to store:

a performance benchmark relating to one or more of the alternative technology products of each technology component, wherein the performance benchmark relating to an alternative technology product includes a level of performance provided by the technology product corresponding to at least one performance metric;

the at least one performance metric relates to load performance or processing speed performance.

5. The system of claim **4**, wherein the at least one processor is further configured to:

receive a request for a level of performance associated with a technology component;

select a technology product for the technology component that satisfies the requested level of performance, based on the performance benchmark of the technology product; and

build a model application environment using the technology product for the technology component.

6. The system of claim **1**, wherein the technology components comprise one or more of a middleware product, an operating system, an implementation technology, a cloud provider, a network technology, a virtualization technique and a server less computing technology.

7. The system of claim **1**, wherein the at least one processor is further configured to:

detect that a new technology product has been added to one of the technology components; and

build at least one additional model application environment having the new technology product selected for the technology component.

8. The system of claim **1**, wherein the at least one processor is further configured to:

run the software application in the current application environment and each of the model application environments at a plurality of loads, wherein each load includes a pre-selected number of logins and a pre-selected amount of test data; and

collect the plurality of performance metrics for each of the loads.

10. The system of claim **1**, wherein the plurality of performance metrics comprises metrics relating to one or more of application server performance, type of data handled by the software application, database performance, security analysis, input and output analysis, deployment on cloud platforms.

11. The system of claim **1**, wherein the at least one processor is configured to:

segregate the performance metrics collected for the current application environment and each of the model application environments;

compare the current application environment and the model application environments based on each performance metric collected for the respective application environments; and

determine the recommendation of the different technology product for the at least one technology component based on the comparison.

12. The system of claim **11**, wherein the at least one processor is further configured to:

determine based on the comparison a recommendation to use at least one of the model application environments for the software application, wherein the at least one model application environment comprises the technology product selected for the at least one technology component; and

adding the recommendation to use the at least one model application environment in the recommendation report.

13. The system of claim **12**, wherein the at least one processor is configured to determine the recommendation to use the at least one model application environment by:

assigning a score to each of the performance metrics collected for the current application environment and each of the model application environments, wherein a higher score is assigned to a higher level of performance related to the performance metric;

comparing the respective scores assigned to each performance metric collected for the current application environment and each of the model application environments;

detecting that the at least one model application environment has a higher score associated with at least one performance metric as compared to the respective score of the at least one performance metric collected for the current application environment; and

determining the recommendation to use the at least one model application environment based on the detecting.

14. The system of claim **11**, wherein the at least one processor is further configured to:

sort the performance metrics based on one or more requirements of the software application; and

select as the different technology product, a technology product that is available for the at least one technology component and satisfies at least one requirement of the one or more requirements of the software application.

15. A method for processing an application environment of a software application, comprising:

obtaining information relating to a current application environment being used to run a software application, wherein the current application environment comprises one or more technology components including one or more hardware technology components, one or more software technology components or a combination thereof, wherein each of the technology components uses a technology product from a plurality of alternative technology products available for the technology component;

building, based on the obtained information, a plurality of model application environments for the software application, wherein each of the model application environments comprises a different combination of the technology products used for the technology components;

running the software application using the current application environment and each of the model application environments in a simulated environment;

collecting a plurality of performance metrics related to performance of the software application in the current application environment and each of the model application environments while running in the simulated environment; and

generating a recommendation report based on the performance metrics, wherein the recommendation report comprises a recommendation of a different technology product for at least one of the technology components used in the current application environment, wherein the different technology product is different from a current technology product used for the at least one technology component in the current application environment; and

16. The method of claim **15**, wherein the information relating to the current application environment comprises information relating to the one or more technology components of the current application environment and the current technology product used for each of the one or more technology components.

17. The method of claim **16**, wherein building the plurality of model application environments comprises:

identifying the plurality of alternative technology products for each of the one or more technology components, wherein each of the plurality of alternative technology products is different from the current technology product of the technology component; and

building each of the plurality of model application environments by selecting for at least one of the technology components a technology product from the plurality of alternative technology products such that the model application environment uses a different combination of the technology products that is different from combinations of the technology products selected for other model application environments.

18. A computer-readable medium for processing an application environment of a software application, the computer-readable medium storing instructions which when executed by a processor perform a method comprising:

obtaining information relating to a current application environment being used to run a software application, wherein the current application environment comprises one or more technology components including one or more hardware technology components, one or more software technology components or a combination thereof, wherein each of the technology components uses a technology product from a plurality of alternative technology products available for the technology component;

building, based on the obtained information, a plurality of model application environments for the software application, wherein each of the model application environments comprises a different combination of the technology products used for the technology components;

running the software application using the current application environment and each of the model application environments in a simulated environment;

collecting a plurality of performance metrics related to performance of the software application in the current application environment and each of the model application environments while running in the simulated environment; and

generating a recommendation report based on the performance metrics, wherein the recommendation report comprises a recommendation of a different technology product for at least one of the technology components used in the current application environment, wherein the different technology product is different from a current technology product used for the at least one technology component in the current application environment; and

19. The computer-readable medium of claim **18**, wherein the information relating to the current application environment comprises information relating to the one or more technology components of the current application environment and the current technology product used for each of the one or more technology components.

20. The computer-readable medium of claim **19**, wherein building the plurality of model application environments comprises:

identifying the plurality of alternative technology products for each of the one or more technology components, wherein each of the plurality of alternative technology products is different from the current technology product of the technology component; and

building each of the plurality of model application environments by selecting for at least one of the technology components a technology product from the plurality of alternative technology products such that the model application environment uses a different combination of the technology products that is different from combinations of the technology products selected for other model application environments.

* * * * *